IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 26, 2018, accepted November 7, 2018, date of publication November 21, 2018,
date of current version December 27, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2882580

Wait Time Prediction for Airport Taxis Using Weighted Nearest Neighbor Regression

MOHAMMAD SAIEDUR RAHAMAN, YONGLI REN, MARGARET HAMILTON, AND FLORA D. SALIM

School of Science, Computer Science & Information Technology, RMIT University, Melbourne, VIC 3000, Australia

Corresponding author: Mohammad Saiedur Rahaman (saiedur.rahaman@rmit.edu.au)

This work was supported by the Australian Research Council's Linkage Projects Funding Scheme through the project Integrated Smart Airport Services under Project LP120200305.

ABSTRACT In this paper, we address the neighborhood identification problem in the presence of a large number of heterogeneous contextual features. We formulate our research as a problem of queue wait time prediction for taxi drivers at airports and investigate heterogeneous factors related to time, weather, flight arrivals, and taxi trips. The neighborhood-based methods have been applied to this type of problem previously. However, the failure to capture the relevant heterogeneous contextual factors and their weights during the calculation of neighborhoods can make existing methods ineffective. Specifically, a driver intelligence-biased weighting scheme is introduced to estimate the importance of each contextual factor that utilizes taxi drivers' intelligent moves. We argue that the quality of the identified neighborhood is significantly improved by considering the relevant heterogeneous contextual factors, thus boosting the prediction performance (i.e., mean prediction error < 0.09 and median prediction error < 0.06). To support our claim, we generated an airport taxi wait time dataset for the John F. Kennedy International Airport by fusing three real-world contextual datasets, including taxi trip logs, passenger wait times, and weather conditions. Our experimental results demonstrate that the presence of heterogeneous contextual features and the drivers' intelligence-biased weighting scheme significantly outperform the baseline approaches for predicting taxi driver queue wait times.

INDEX TERMS Heterogeneous contextual features, neighborhood identification, wait time prediction, feature weighting.

I. INTRODUCTION

Taxis are regarded as the most convenient mode of transport for transfer between the airport and the city. Airport satisfaction ratings depend on the proper management of both taxi and passenger queues. Aiming to maintain a demand-supply equilibrium of taxis, the airport transport managers employ an approach where it requires extensive human intervention. Figure 1 presents a schematic overview of a traditional airport management system used to manage taxi and passenger queues. We can see that the queue managers continuously monitor the concurrent queues related to taxis and passengers and instruct taxi drivers to join the passengers at the terminal when there is demand. To ensure the seamless operation of this process, the queue manager estimates the demand for taxis in future. Request for more taxis is sent to the taxi fleet managers who then send the requested number of taxis to meet the future demand or an instant shortage of taxis at the airport taxi rank. However, the human error in manual taxi demand estimation causes taxi drivers to experience unexpected wait times at the airport taxi rank while waiting for the passengers. Moreover, long queues of taxis cause traffic congestion and wasted land use. This also influences taxi drivers not to make an airport trip. Consequently, it also causes long queue wait times for the passengers. Therefore, it is very important to provide effective prediction of the taxi queue wait times at airport taxi rank since it can help airport queue managers to plan timely taxi and passenger queue management.

Queue wait time prediction is a challenging problem which is highly affected by many heterogeneous contexts including the dynamic nature weather conditions, taxi, passenger and flight arrivals [1]. Taxi drivers experience different wait times while waiting in the taxi rank. Figure 2 (top) shows a comparison of average queue wait times for two different Mondays

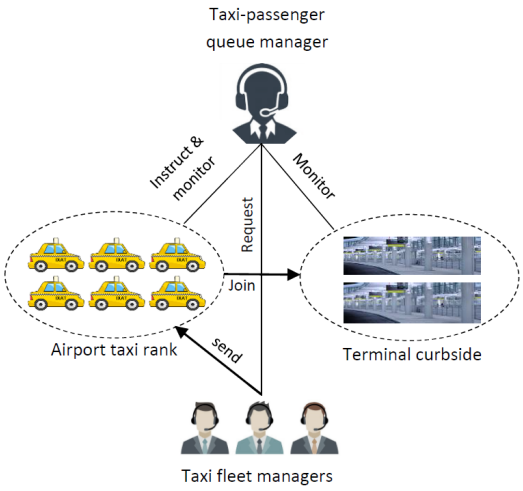

FIGURE 1. Manual taxi-passenger queue management system.

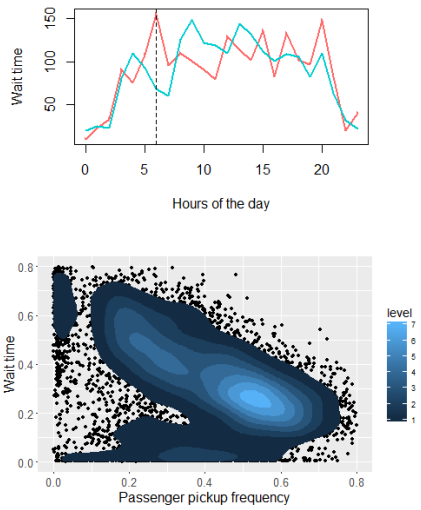

FIGURE 2. Variation in taxi waiting times on two Mondays over two different weeks of May 2013 (top). The density map (bottom) shows the variation of wait times with respect to current passenger pickup frequency.

of April 2013 at the central taxi holding area of JFK international airport in New York City. We can see that the wait times vary throughout the day. The variations in densities of the wait times with respect to taxi passenger pickup frequency are well scattered as can be seen from the two- dimensional density plot in Figure 2 (bottom). The highly dense area relates to where the pickup frequency is high and may result in low queue wait times. It is not sufficient to conclude that a low pickup frequency can always result in higher wait times as some of the highlighted areas relate to low passenger pickups. This also indicates that it is a complex problem to identify the importance of other related contextual factors for taxi driver queue wait time prediction.

Queue wait time prediction has been studied in many application areas [2]. This problem can be addressed using various machine learning techniques [3]. The neighborhood-based method such as k-Nearest Neighbor (k-NN) regression has shown its effectiveness in recent research of wait time prediction [4]. The general idea of k-NN regression is to predict a real valued response (i.e. target score) by taking the weighted average over neighboring response values. We choose to use the neighborhood-based method in this paper to address our problem because of its simple implementation and performance guarantees [5]. However, the prediction performance of neighborhood-based methods (i.e. k-NN regression) depends on the proper identification of neighbors (i.e. neighborhood) and relevant features [6]. Ren *et al.* [7] found that an improvement in terms of the quality of identified neighborhood can further improve the prediction accuracy. In this paper, we formulate the neighborhood identification problem as the queue wait time prediction problem at the airport. We aim to investigate the effects of heterogeneous external contexts in prediction performance, which may have direct influence on the queue wait time. For example, the bad weather may cause high demand for taxis, and the delayed flight arrivals may cause long taxi queues. Moreover, the dynamic occurrence and heterogeneous nature of these factors make the taxi queue wait time prediction at the airport a complex task. We argue that the computation and consideration of the influence of these factors are vital for the quality neighborhood identification and thus for taxi queue wait time prediction. In this paper, we aim at the following challenging problem:

How to identify a dense quality neighborhood for k-NN based regression methods to predict taxi queue wait time by considering heterogeneous contextual features, e.g. time, weather, flight information and taxi trips?

We first build a large-scale taxi queue wait time dataset for JFK international airport in New York City. The dataset includes hourly taxi queue wait time and different heterogeneous contextual information which were obtained by fusing taxi trip logs, airport passenger wait times and weather conditions data. Then, we provide a comprehensive analysis of the relationship between the heterogeneous features and the taxi queue wait times. Based on the analysis, we propose a driver intelligence-biased feature weighting scheme to identify a dense quality neighborhood for predicting taxi queue wait time using k-NN regression. The experiment results indicate that the relevant weighted heterogeneous contextual features can significantly improve the quality of the identified neighborhood, thus significantly boosting the prediction performance. The contributions of this paper are as follows:

- Fusion of three real-world contextual datasets for taxi queue wait time prediction and extraction of heterogeneous features;
- Feature selection and analysis of heterogeneous features for taxi queue wait time prediction;
- A driver intelligence-biased feature weighting scheme to identify a dense high quality neighborhood for taxi queue wait time prediction using k-NN based regression methods;

This paper is organized as follows: Section 2 reviews related literature. Section 3 describes dataset preparation, feature extraction and pattern analysis from the three real-world datasets. Section 4 describes a model to identify dense neighborhood for taxi queue wait time prediction using k-NN regression. In Section 5, we show experimental results and comparison with the state-of-the-art approach. Finally, Section 6 concludes the paper.

II. RELATED WORK

Both taxi drivers and passengers can experience long queue wait times at airports [8]. The decisions of taxi drivers to make a trip to/from the airport directly affects the occurrence of these queues [1] which are also influenced by many external factors such as weather, flight arrivals and time. Accurate estimations of the taxi queue wait times over the course of the day can encourage taxi drivers to make more airport trips and help manage the demand-supply of taxis. However, it is challenging to find relevant factors with their respective importance levels to be considered for queue wait time estimation.

Research projects have been conducted to investigate the airport taxicab demand-supply equilibrium [9], [10]. There are several research projects that model and analyze factors related to taxi drivers' next passenger pickup decisions [11]–[14]. The uncertainty of the queue formation along with the limitation in manual taxi demand estimation play important roles towards taxi drivers' decisions about making an airport trip or not [12]. In [1], a regression model is used to model the taxi driver's next pickup decision for an airport trip. The model uses a binary decision of ''airport pick-up'' or ''cruising for customers'' at the end of each trip. Another passenger search model for taxi drivers is presented in [15] to investigate the hourly changes in decisions of vacant taxi drivers while searching for passengers during a day. To model taxi drivers' customer-search behaviors, a two-layer decision framework is proposed in [16]. The model considers location choice and route choice behavior and uses GPS trajectories. The trajectory data also has been used to analyze taxi driver's route choice behavior [17], to predict KPIs of on-demand transport services [18] and to monitor congested corridors in busy smart cities [19]. In [20], spatio-temporal taxi traveling patterns are studied. Specifically, the patterns during the battle between two Chinese taxi booking mobile apps are taken into consideration for analyzes. An optimal route searching solution for taxi drivers is proposed in [21] which is based on the information of the traffic dynamics. Other analyzes of taxis mobility patterns are also presented in [22] and [23]. In [24], a prediction model is proposed to estimate the taxi and passenger demand. The research works presented in [25] and [26] predict queue contexts related to taxis and passengers at airports where contexts are four categorical labels. There are research papers that consider wait time as an additional feature for modeling taxi drivers' next pickup decisions. However, wait time prediction is an important and complex task which still requires special attention.

There are several papers which discuss research conducted for predicting and modelling wait time in different applications [3], [4], [27]–[29]. Wu *et al.* [30] have developed a system which continuously monitors each taxi stand and takes account of the numbers of taxis queuing and passing the taxi stand, as well as the traffic conditions in the area around the stand. Zhang *et al.* [3] have recommended sensing the fuel consumption of taxi drivers with a view to minimizing queue times at petrol stations. On the passenger side, Anwar *et al.* [9] have considered passenger movement through an airport, with a view to sending taxis this information for the demand so that they can service the longest queue first with short wait times.

The nearest neighbor regression (k-NN regression) is an effective machine learning approach for prediction on a numeric scale in various applications. A shop queue wait time prediction using k-NN regression is presented in [4] which utilizes three temporal contextual features. However, neighborhood identification is still a challenging area for nearest neighbor regression. Various approaches have been adopted such as distance weighting [31], [32] of nearest neighbors. Feature weighting has shown its effectiveness in regard to increased prediction performance in many application domains [33]–[38]. In [4], feature importance score is calculated by building a linear regression model. This approach is effective with incomplete and non-uniform data. Another feature weighted distance measure for k-NN is proposed in [39]. It is based on the mutual information between a feature and the class value. The mutual neighborhood information is used to boost the performance of nearest neighbor classification in [40]. The effectiveness of feature weighting during neighborhood calculation of k-NN regression is still unexplored. The nearest neighbor regression has been used effectively in real-world wait time prediction applications with a small number of features [4]. Taxi queue wait time prediction at the airport using nearest neighbor regression is a challenging issue because of the presence of various contexts (i.e. weather, flight arrivals, flight processing, time). Since these contexts are heterogeneous in nature, the computation of relationships between these contexts and the queue wait time is a complex task.

Research has shown that the use of expert knowledge is able to increase the prediction accuracy [41]. In [42], a categorization framework is proposed which refers to the use of domain-specific information for feature weighting. In our problem of queue wait time prediction, it is difficult to utilize experts for feature weighting. We leverage the idea that the expert drivers use their expertise to go to a place for passenger pickup rather than cruising randomly [43]. We take this note and consider the taxi drivers' intelligent moves for feature weight computation. Then the computed feature weights are used to find the quality neighborhood and to predict the taxi queue wait time using k-NN regression. To the best of our knowledge, this is the first work in this direction.

III. DATASET DESIGN AND CONTEXTUAL ANALYSIS

To prepare the airport taxi driver queue wait time dataset, we fuse three real-world datasets from New York city: the taxi trip data, the airport passenger wait times data and the weather condition data.

The JFK international airport is one of the busiest airports in the U.S. and the busiest in New York City in terms of flight processing and passenger handling. The taxi rank at the JFK is far away from the passenger terminals. Taxis need to enter this rank area and wait before picking a passenger up. The taxi dispatch manager at the JFK dispatches taxis from this taxi rank based on the demand in different passenger terminals [12]. Another airport in New York City is the LaGuardia airport. This airport is not covered by the border control facility and hence no passenger arrival and wait time information is available. Therefore, we choose the JFK international airport as our case location to prepare our dataset of taxi driver queue wait times at the airport.

A. TAXI TRIP LOGS

The first dataset is the real-world taxi trip dataset from New York City known as NYC taxi trip data.[1] This dataset is collected by the New York City Taxi & Limousine Commission. Everyday in New York, 13,000 taxis generate 500,000 trips on average totaling 175 million trips per year. In this dataset, each record is equivalent to one taxi trip. A trip is described by its start and end geo-locations with corresponding time-stamps, the distance of the trip, the fare and tip amount of the trip, total number of on board passengers during the trip and a medallion, which is the unique identification number of the taxi.

B. PASSENGER DATA

Airport passenger data is available from the public U.S. Customs and Boarder Protection website.[2] This dataset provides information about hourly passenger wait times at different U.S. airports. Additional features include hourly numbers of flights and passenger arrivals with the number of passengers processed at the passenger processing booths.

C. WEATHER DATA

We collected weather data from the Weather Underground website.[3] This dataset contains hourly weather information including temperature, dew point, wind speed, precipitation, weather condition (e.g., clear, overcast, mostly cloudy) and weather events (e.g., normal, rain, snow, rain-snow).

D. DATA FUSION AND FEATURE EXTRACTION

Here, we fuse the NYC taxi trip data, the passenger data and the weather data, and extract corresponding features:

- *NYC taxi trip data:* hourly queue wait time for taxis, passenger pickup frequency, passenger drop-off frequency, frequency of trips that start from the airport after an airport drop-off.
- *Passenger data:* hourly frequency of flight arrivals, passenger arrivals, passenger wait times at the passenger processing booth, number of passenger processing booth.
- *Weather data:* hourly temperature, precipitation, wind speed, dew point, weather condition and weather events.

For all these features, we compute the feature values in the previous, current and next hour. We fuse the features extracted from the three real-world datasets for our experiment and analysis. In the final dataset, each record describes an hourly time stamp totaling 8760 data points during the year of 2013. Additionally, we consider features such as the hour of day, the day of week, and the week number of the year.

E. CONTEXTUAL ANALYSIS

We analyze the taxi queue wait time from the perspective of *time*. We examine the hourly patterns of the taxi queue wait times, taxi passenger pickup and drop-off frequencies, passenger arrivals, and passenger wait times. Figure 3 shows the hourly patterns and the Empirical Cumulative Distribution Function (ECDF) of taxi queue wait times. The ECDF in Figure 3(a) shows that 80% of the data points have wait times less than or equal to 90 minutes. Figure 3(b) shows the hourly patterns for normalized taxi queue wait times. We can see that the highest wait times are seen during 04:00 and 09:00. Figure 3(c) and (d) show the hourly patterns in passenger pickup and drop-off by taxis. A spike is seen at 15:00 for passenger drop-off and after that it reduces till mid-night. The reason may be due to the high volume of departing flights in the afternoon. Two clear spikes can be seen for passenger pickup frequencies: one at 07:00 in the morning while the other starts at 15:00 and continues until mid-night. Figure 3(e) illustrates the hourly frequency of taxi trips from JFK starting after a passenger drop-off at JFK. We can see that a large number of taxis decide to pickup their next passenger from the airport between 13:00 and 19:00. Figure 3(f) shows the density of taxis where they leave the airport after a passenger drop-off. The hours between 01:00 and 03:00 share the majority of the densities. This is expected because these hours are the off-peak hours at JFK. We also can see from Figure 3(g) that the passenger arrivals maintain a similar trend line to the passenger pickup frequency. However, a difference of an hour is seen in the rise and fall of the spikes. Figure 3(h) shows the hourly average passenger wait times. We can see near smooth trend in wait times throughout the day except for one sharp rise from 03:00 when the flights start to arrive.

In Table 1, we present all the correlation statistics between the hourly taxi queue wait times and all other features extracted from three heterogeneous contextual datasets including *passenger*, *trip*, and *weather*. We compute the Pearson's correlation coefficient to measure the relationship between wait times and the other features. Specifically, a negative correlation is found with total flight arrivals in the

[1] http://www.andresmh.com/nyctaxitrips/
[2] http://awt.cbp.gov/
[3] https://www.wunderground.com/

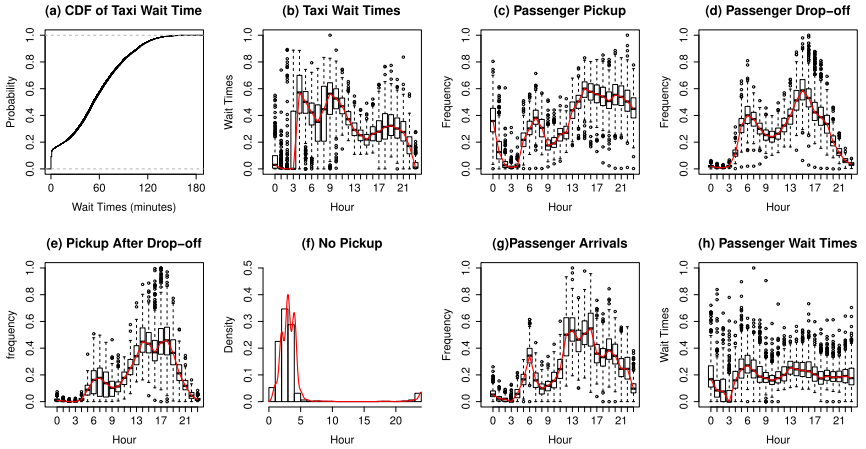

FIGURE 3. Contextual Analysis: (a) Cumulative distribution function of taxi wait times and (b)-(h) hourly patterns of various contexts.

TABLE 1. Pearson's correlations between average queue wait times and all the contextual features.

Contexts	Features	Corr.
Passenger	Total passenger in previous hour	-0.26
	Total passenger in current hour	-0.09
	Total passenger in next hour	+0.13
	Total flights in previous hour	-0.30
	Total flights in current hour	-0.13
	Total flights in next hour	+0.10
	Total booths in previous hour	-0.23
	Total booths in current hour	-0.05
	Total booths in next hour	+0.16
	Average passenger waiting in previous hour	-0.15
	Average passenger waiting in current hour	-0.06
	Average passenger waiting in next hour	+0.12
Trip	Passenger pickup frequency in previous hour	-0.49
	Passenger pickup frequency in current hour	-0.44
	Passenger pickup frequency in next hour	-0.21
	Passenger drop-off frequency in current hour	+0.12
	Drop and pick frequency in current hour	-0.07
Weather	Temperature (°C) in previous hour	-0.02
	Temperature (°C) in current hour	+0.01
	Temperature (°C) in next hour	+0.03
	Dew point in previous hour	-0.01
	Dew Point in current hour	-0.01
	Dew Point in next hour	-0.01
	Wind speed(Kmph) in previous hour	-0.03
	Wind speed(Kmph) in current hour	-0.01
	Wind speed(Kmph) in next hour	+0.02
	Precipitation(mm) in previous hour	-0.05
	Precipitation(mm) in current hour	-0.04
	Precipitation(mm) in next hour	-0.04
	Snow in current hour	+0.03
	Rain in current hour	-0.08

previous hour, total passenger arrivals in the previous hour, total flight processing booths in the previous hour, passenger pickup frequency in the previous hour and passenger pickup frequency in the current hour. This implies that the more flights, passengers, passenger processing booths and passenger pickup frequencies, the shorter the taxi queue wait time.

IV. QUEUE WAIT TIME PREDICTION

In this section, we define the problem of taxi queue wait time prediction and formulate the k-NN based regression

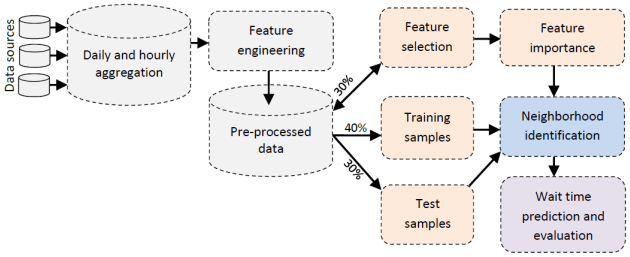

FIGURE 4. Overview of neighborhood-based taxi queue wait time prediction system.

methods for this scenario. Specifically, we introduce a feature weighting scheme to improve the quality of identified neighborhood, so as to improve the accuracy of queue wait time prediction. We also present the overview of our entire taxi queue wait time prediction system. The system begins by integrating data from various heterogeneous data sources. A daily aggregation of the fused data is followed by an hourly aggregation to ensure the same time frame is considered for all the extracted features. Additional features are calculated in the feature engineering stage which completes the data pre-processing tasks. The pre-processed data is sliced into three parts. The first part is used to select relevant features. Based on the selected features, the second and third parts of the data are filtered and kept for training and testing the prediction process. The first part of data with selected features is also utilized for feature weight calculation. These weight scores are used during the distance calculation of neighborhood-based methods. Specifically, a test sample is taken to measure the distance from all the training samples since the neighborhood-based methods do not require any explicit training phase. Finally, the system evaluates the prediction results which can be used to ensure the effectiveness of any decision. Figure 4 shows an overarching view of our taxi queue wait time prediction system. In Table 2, we list the notations used in this paper.

TABLE 2. List of notations.

Notation	Description of
T_i	i^{th} instance in the training dataset.
T_Q	An hourly query time window.
$\overline{w}(T_i)$	Actual wait time during time window T_i.
$\overline{\overline{w}}(T_Q)$	Predicted wait time during T_Q
F_c	Set of contextual features.
$d(T_Q, T_i)$	Distance between T_Q and T_i.
a_j	The j^{th} contextual feature.
ω_j	Weight of j^{th} feature.
DI	Driver intelligence.
MI	Mutual information.
$I(a_j; \overline{w}(T_i)\|DI)$	Driver intelligence-biased MI.

TABLE 3. Statistically significant features computed from time contexts and the three heterogeneous contextual datasets, including *passenger*, *trip*, and *weather*.

Contexts	Features	95%CI
Time	Day of the week	(-0.00593, -0.00241)
	Hour of the day	(0.005447, 0.006999)
	Week number of the year	(-0.00099, -0.00049)
Passenger	Total passenger in current hour	(-0.12461, -0.06930)
	Total passenger in next hour	(0.012649, 0.06580)
	Average passenger waiting in current hour	(-0.22139, -0.13371)
	Average passenger waiting in next hour	(0.019088, 0.105692)
Trip	Passenger pickup frequency in previous hour	(-0.32382, -0.25338)
	Passenger pickup frequency in current hour	(-0.46570, -0.38555)
	Passenger drop-off frequency in current hour	(0.265628, 0.344074)
Weather	Temperature in previous hour	(-0.68603, -0.45740)
	Temperature in next hour	(0.488688, 0.714952)
	Precipitation in previous hour	(-0.34068, -0.11975)
	Precipitation in next hour	(-0.24090, -0.01995)

A. PROBLEM DEFINITION

Let, T_i be the i^{th} instance in the training dataset that represents any hourly time window, where, $i = 1, 2, 3, ..., 8760$. In our taxi queue wait time dataset, each instance T_i is described by a set of contextual features F_c. The hourly taxi queue wait time during T_i is denoted as $\overline{w}(T_i)$ which is the taxi drivers' average time spent in the queue from the time of arrival during T_i until the next passenger pickup. Given a training dataset, a query instance (T_Q) and its corresponding contextual features F_c, we predict the hourly taxi queue wait time as: $(T_Q, F_c) \rightarrow \overline{\overline{w}}(T_Q)$. We aim to find the quality dense neighborhood for nearest neighbor regression.

B. FEATURE SELECTION

Not all the extracted features are important for the queue wait time prediction. Therefore, we perform a feature selection to enhance the prediction performance. For this purpose, we build a multiple regression model to predict the queue wait times. If $\hat{Y}$ is the target score, we write the multiple regression model using n number of features as follows:

$$\hat{Y} = \beta_1 a_1 + \beta_2 a_2 + \beta_3 a_3 + ... + \beta_n a_n \tag{1}$$

Here, a_j is the j^{th} feature and β_j is corresponding feature coefficient of a_j; $j = 1, 2, 3, ..., n$. Then we investigate the importance of each feature to the research problem. For simplicity, we randomly select a subset of $n = 15$ features and build the regression model. Then we examine the coefficients of all these features within the model by using a 95% confidence interval. Specifically, we examine if 0 is within this interval. If so, it indicates that the coefficient can have a value of 0 thus the feature has no or less effect to predict the target score (i.e. queue wait time). We consider such features as unimportant. In every iteration, we leave one unimportant feature out and include a new one for the next round. In final dataset, we find a total of 14 features which are significantly associated with the queue wait times. We consider the features from the *time* context as well along with *passenger*, *trip*, and *weather*. The statistically significant features from four different contexts are listed in Table 3.

C. DRIVER INTELLIGENCE-BIASED WEIGHTING

In this section, we calculate feature weights for queue wait time prediction. The recent literature [43] shows that the experienced drivers do no prefer random cruising for passengers after a passenger drop-off. Instead, they usually go to the place they know well for picking up the next passengers. We assume that this is also applicable in our scenario. Motivated by this fact, we consider the hourly frequency of taxi trips that are initiated from the airport after a precedent passenger drop-off at the airport. We call this frequency the Drivers' Intelligence (DI).

The mutual information is a measure of the mutual dependence between two random variables. Therefore, we can calculate the amount of mutual dependence between the queue wait time and all other features available in our queue wait time dataset after feature selection. Specifically, we calculate the conditional mutual information by putting the DI as a condition. Henceforth, we refer to this conditional mutual information as the DI-biased mutual information. We calculate the DI-biased mutual information ($I(a_j; \overline{w}(T_i)|DI)$) by adapting the universal equation for computing conditional mutual information as follows:

$$-\sum_{a_j, \overline{w}(T_i), DI} p(a_j, \overline{w}(T_i)) \log \frac{p(a_j, \overline{w}(T_i)|DI)}{p(a_j|DI)p(\overline{w}(T_i)|DI)}, \tag{2}$$

In Eq. 2, a_j is any contextual feature; $\overline{w}(T_i)$ is the target score (i.e. queue wait time) and p denotes the probability. Table 4 lists the corresponding DI-biased mutual information scores for different features. In the next step, we normalize these scores of DI-biased mutual information between 0 and 1 and used as DI-biased feature weights ω_j. Here, ω_j is the feature weight of j^{th} feature.

D. FORMULATION OF k-NN METHODS

Given a set of training samples, we formulate the problem of predicting a corresponding target score using k-NN regression. Let each sample T_i in the training data be described by a d-dimensional vector of contextual features and a target

TABLE 4. D)-biased mutual information.

Contexts	Features	$I(a_j; \overline{w}(T_i)\|DI)$
Time	Day of the week	0.050
	Hour of the day	0.829
	Week number of the year	0.094
Passenger	Total passenger in current hour	0.400
	Total passenger in next hour	0.348
	Average passenger waiting in current hour	0.108
	Average passenger waiting in next hour	0.109
Trip	Passenger pickup frequency in previous hour	0.228
	Passenger pickup frequency in current hour	0.385
	Passenger drop-off frequency in current hour	0.696
Weather	Temperature(°C) in previous hour	0.058
	Temperature(°C) in next hour	0.049
	Precipitation(mm) in previous hour	0.003
	Precipitation(mm) in current hour	0.004

score $\overline{w}(T_i)$ as follows:

$$\langle a_1(T_i), a_2(T_i), a_3(T_i), ..., a_d(T_i), \overline{w}(T_i)\rangle,$$

To predict the target score of query instance T_Q, the distances between T_Q and all the training samples T_i are calculated as follows:

$$d(T_Q, T_i) = \sqrt{\sum_{j=1}^{d} [a_j(T_Q) - a_j(T_i)]^2}, \tag{3}$$

Here, $a_j \in F_c$ is the j^{th} contextual feature of T_i and T_Q; $j = 1, 2, 3, ..., d$. Note that the basic k-NN regression treats each feature equally during this distance calculation. However, the contribution of each feature can be taken into account by rewriting Eq. 3 as follows:

$$d(T_Q, T_i) = \sqrt{\sum_{j=1}^{d} \omega_j * [a_j(T_Q) - a_j(T_i)]^2}, \tag{4}$$

Here, ω_j is the weight of j^{th} feature. Next, the k-nearest neighbors of T_Q are identified by sorting the values of $d(T_Q, T_i)$ in ascending order. Let, $T^{NN} = \{T_1^{NN}, T_2^{NN}, T_3^{NN}, ..., T_k^{NN}\}$ be the set of k-nearest neighbors of T_Q. If $\overline{w}(T_i^{NN})$ is the target score of T_i^{NN}, the predicted target score $\overline{\overline{w}}(T_Q)$ of the query instance T_Q is calculated by averaging the target scores of k-Nearest Neighbors as follows:

$$\overline{\overline{w}}(T_Q) = \sum_{i=1}^{k} \overline{w}(T_i^{NN})/k, \tag{5}$$

It should be noted that the key here is to estimate the weights in Eq. 4 for each feature appropriately so as to get a better neighborhood for higher prediction accuracy.

V. EXPERIMENTS AND RESULTS

We design two sets of experiments to evaluate the effectiveness of the proposed DI-biased feature weighting scheme:

1) Taxi Queue Wait Time Prediction: We predict the taxi queue wait time and compare with different weighting methods as follows:

- *Baseline* [4]: The baseline nearest neighbor estimation approach employs a regression based optimization for feature weighting. It considers only three attributes for queue wait time prediction (time of the day, the day of the week, and week number of the year), and weights the features based on the coefficients obtained from a linear regression model.
- *LR-trained weights*: Instead of only three, all 14 significant contextual features from Table 3 are considered. The feature weights are calculated by normalizing the coefficients obtained from a trained linear regression (LR) model which are to be used for the nearest neighbor estimation.
- *Equal weights*: In this approach, all 14 significant contextual features from Table 3 are considered with equal weights for the nearest neighbor estimation.
- *MI-based Weights*: The Mutual Information (MI)-based weights includes all the significant contextual features from Table 3. The pure MI between each feature and the target (i.e. taxi queue wait time) is calculated, normalized between 0 and 1, and used as feature weights for nearest neighbor estimation.
- *DI-biased weights*: The DI-biased feature weighting is the proposed weighting scheme of contextual features. The scores for each feature obtained from Eq. 2 are normalized to be used as feature weights for nearest neighbor estimation.

2) Neighborhood Density/Quality: we evaluate and compare the density and quality of the neighborhood between the baseline and our proposed approach.

We use a random 30-40-30 split to conduct our experiments with different feature weighting schemes for Nearest Neighbor Estimation. The first part contains 30% of the samples which were used for feature selection and feature importance calculation. The second part contains 40% of the instances which were used for training purpose, and the third part contains 30% of the instances which were used to test the performance of nearest neighbor estimation. The purpose of using this approach is to minimize the bias of using the entire dataset for feature weighting score calculation. For performance evaluation of queue wait time prediction, we consider the median and mean prediction errors for different k values between 1 and 15. Note that the values of k are chosen arbitrarily.

A. TAXI QUEUE WAIT TIME PREDICTION

Figure 5 and 6 show the comparison of median and the mean prediction errors respectively from all the methods. We can see that the proposed feature weighting method *DI-biased weights* and its three variants (*LR-trained weights*, *Equal weights* and *MI-based weights*) outperform the *baseline* [4] for different k values between 1 and 15 since they produce less prediction errors. Specifically, we can see a median

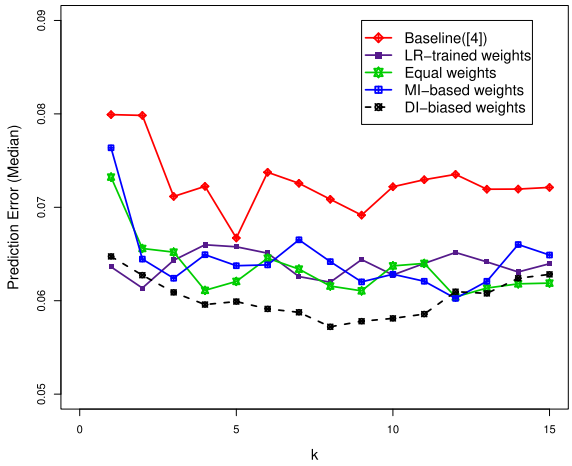

FIGURE 5. Median prediction errors using various feature weighting techniques for varying k-values between 1 and 15.

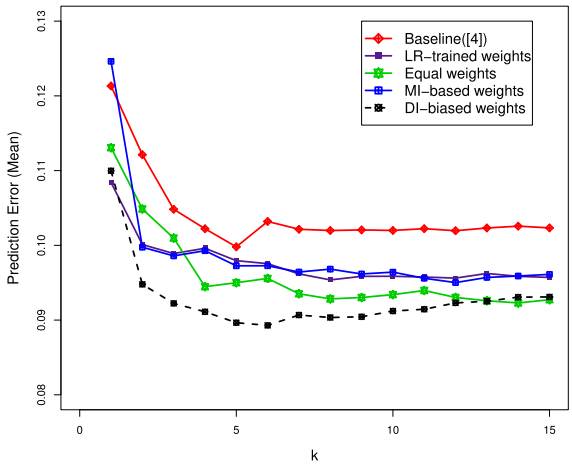

FIGURE 6. Mean prediction errors using various feature weighting techniques for varying k-values between 1 and 15.

prediction error < 0.06 and a mean prediction error < 0.09 for $k = 8$ and 6 respectively.

Next, we examine the statistical significance of this improvement. Specifically, a paired t-test is conducted to examine whether the improvement in prediction errors is significant when comparing with *baseline*. A paired t-test can determine whether the mean differences between two sets of paired samples differs from 0. The mean difference 0 indicates that the paired samples are similar. In our case, the first sample is the prediction errors produced by the *baseline* approach while the second sample is set in turn for the prediction errors produced by the *LR-trained weights*, *Equal weights* and *MI-based weights*. The pair-wise prediction errors are taken into consideration for varying k-values between 1 and 15. Table 5 gives the statistics obtained from the paired t-test. We can see that the improvement in terms of prediction errors is statistically significant between the proposed *DI-biased weighting* method (including its variants) and the *baseline* since the values of t-test statistics (t) differ

TABLE 5. Paired t-test of prediction errors between different feature weighting techniques (Note: we obtained $p < 0.001$ in all cases).

Methods	t-median	t-mean
LR-trained weight *vs* Baseline	08.25	09.04
Equal weights *vs* Baseline	14.46	17.97
MI-based weights *vs* Baseline	09.63	06.54
DI-biased weights *vs* Baseline	17.89	21.41
DI-biased weights *vs* LR-trained weights	05.33	07.21
DI-biased weights *vs* Equal weights	04.39	04.03
DI-biased weights *vs* MI-based weights	05.30	07.55

TABLE 6. Comparison of k-NN regression with DI-biased weighting method and other state-of-the-art methods.

Methods	Mean Prediction Error
Linear Regression	0.121
Multi-layer Perceptron	0.118
Support Vector Regression (SVR)	0.126
Decision Table	0.103
k-NN with DI-biased weight	**0.087**

significantly from 0. The smaller p values (< 0.001) on this small sample size of 15 also supports this significance.

In Table 6, we compare our k-NN with DI-biased weighting technique and other state-of-the-art methods for regression including Linear regression, Multi-layer Perceptron, Support Vector Regression and Decision Table. We can see that k-NN regression with DI-biased weighting method produces a mean prediction error of 0.087 which is no more than the listed state-of-the-art algorithms for taxi queue wait time prediction.

B. NEIGHBORHOOD ANALYSIS

We analyze the ECDFs and the densities of inter-neighbor $NN(p, q)$ distances where $p = 1$ to $14, q = p+1$. We compare *LR-trained weights*, *Equal weights*, *MI-based weights* and *DI-biased weights* with the *baseline*. Figure 7 shows the ECDFs of distances between consecutive neighbors. We can see from the plotted ECDFs that there are some jumps in the ECDFs of inter-neighbor distances when the baseline was applied. This indicates that the subsequent neighbors are not dense enough which results in a sparse neighborhood. On the other hand, the ECDFs for all of the other four approaches show smooth trend lines which imply the existence of a dense neighborhood, among which *DI-biased weights* shows the most smoothness compared to the rest.

We examine the density plots of inter-neighbor distances to supports our claim of achieving the denser neighborhood using DI-biased weighting method. We identify that there is mostly one peak in density plots for *LR-trained weights*, *Equal weights*, *MI-based weights* and *DI-biased weights*. This trend also remains the same when we consider the two furthest neighbors (i.e. 14 and 15). In contrast, we see that there is more than one peak in the first four density plots for the *baseline* approach. As we move towards the furthest nearest neighbors, a long tail in the density distributions can

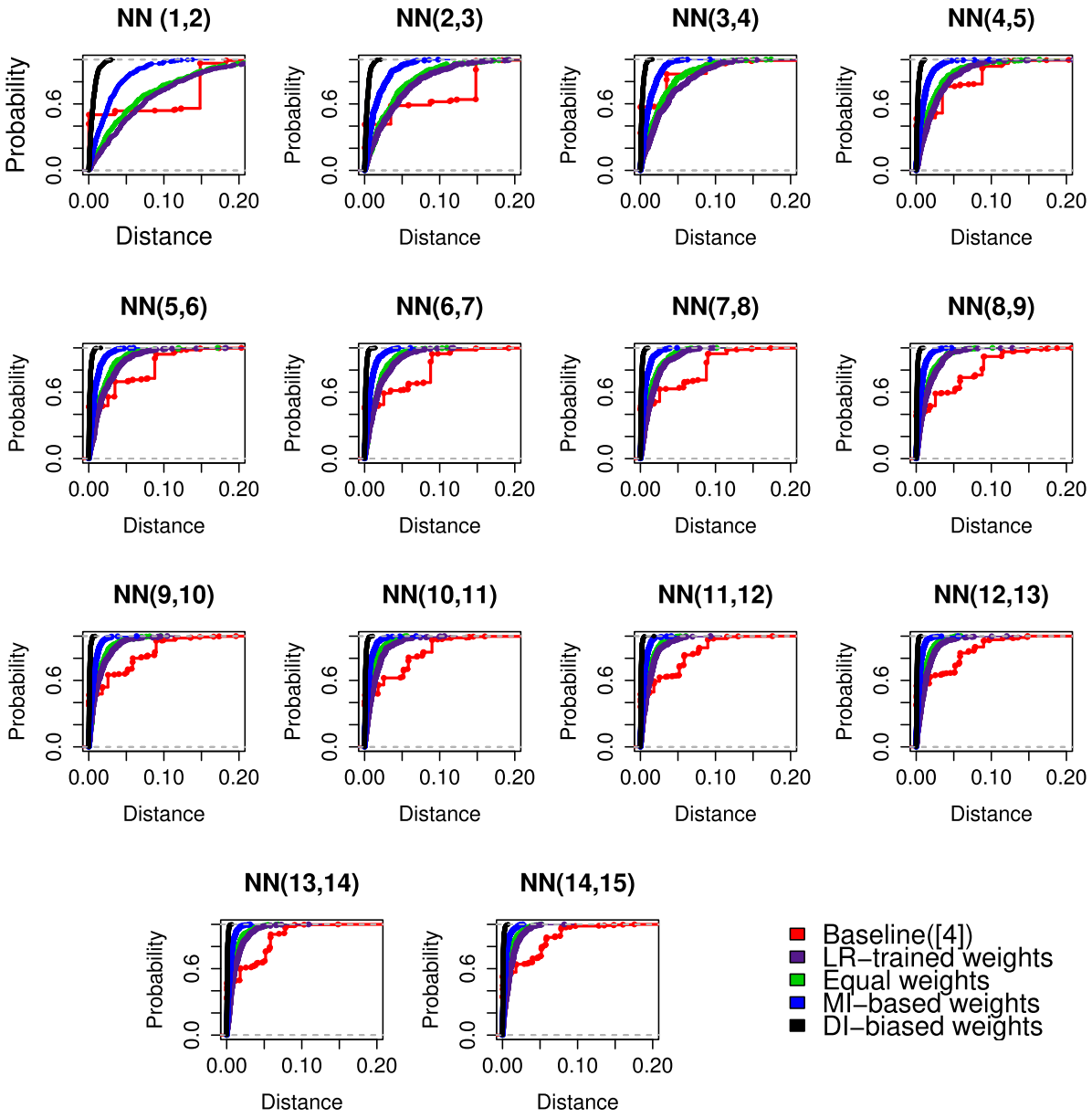

FIGURE 7. ECDFs of inter-neighbor distances using different weighting techniques.

be seen which implies the existence of a sparse neighborhood using the baseline approach. Figure 8 illustrates the density plots of inter-neighbor distances using different weighting techniques.

Next, we analyze the robustness of our *DI-biased weighting* method. Specifically, we examine the changes of inter-neighbor distances with the change in k values. We plot the ECDFs and densities of all the distances between two consecutive neighbors using *DI-biased weights* only.

In Figure 9, we can see from the ECDFs that the first two nearest neighbors are the least dense compared to the next two, and so on. However, as we increase the size of the neighborhood, a more close distance between the two furthest neighbors is seen. This indicates that the *DI-biased weights* is able to find the dense neighborhood when we increase the value of k.

From the density plots in Figure 10, we can see that the furthest neighbors are more dense compared to the closest ones. This implies that the *DI-biased weighting* method shows its robustness in identifying dense neighborhood with any neighborhood size between 1 and 15 in this study with taxi queue wait time dataset.

We examine the relationship between the identified neighborhood and the taxi queue wait time prediction. Specifically, a K-S (Kolmogorov-Smirnov) test is employed. The K-S test measures the difference between ECDFs of distances among identified neighborhoods by applying the *DI-biased weighting* method and *baseline* method respectively in terms

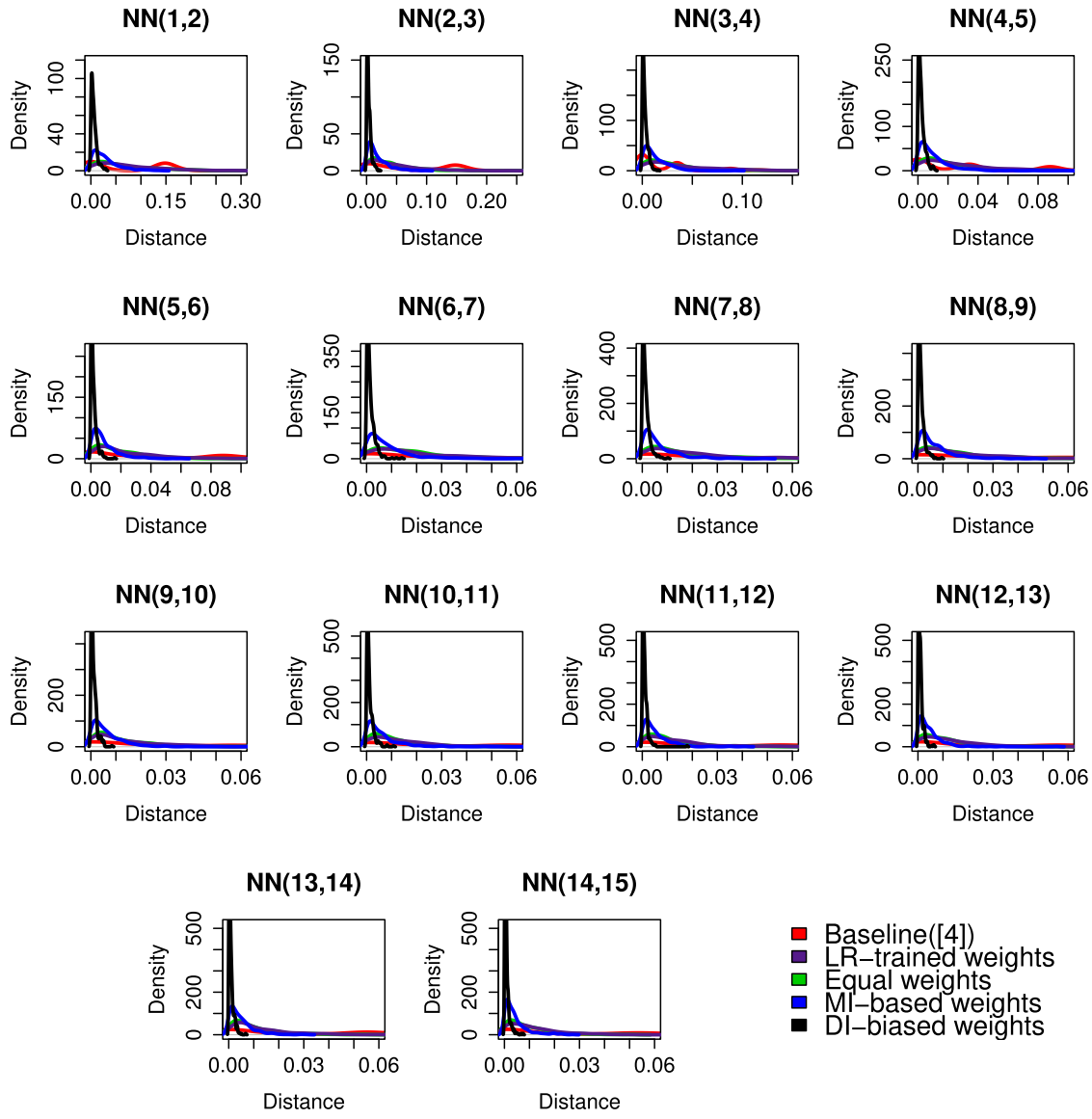

FIGURE 8. Density plots of inter-neighbor distances using different weighting techniques.

TABLE 7. Relationships between the identified neighborhood and the taxi queue wait time prediction errors using K-S (Kolmogorov-Smirnov) test.

$ECDF$s of distances between $NN(p,q)$	D (K-S Test)	p	k	d_error (Median)	d_error (Mean)	Correlation $[D, d\_error(\text{Median})]$	Correlation $[D, d\_error(\text{Mean})]$
(1,2)	0.50	<0.001	2	0.017	0.017		
(2,3)	0.46	<0.001	3	0.010	0.013		
(3,4)	0.55	<0.001	4	0.012	0.011		
(4,5)	0.45	<0.001	5	0.007	0.010		
(5,6)	0.47	<0.001	6	0.015	0.014		
(6,7)	0.46	<0.001	7	0.014	0.011		
(7,8)	0.45	<0.001	8	0.014	0.012	0.484	0.393
(8,9)	0.50	<0.001	9	0.011	0.012		
(9,10)	0.47	<0.001	10	0.014	0.011		
(10,11)	0.47	<0.001	11	0.014	0.011		
(11,12)	0.48	<0.001	12	0.013	0.010		
(12,13)	0.47	<0.001	13	0.011	0.010		
(13,14)	0.48	<0.001	14	0.010	0.009		
(14,15)	0.45	<0.001	15	0.009	0.009		

of D-value which is the maximum difference between these two. We examine the corresponding p-values to show the statistical significance of this difference. In Table. 7, the D-value is around 0.45 and p-value < 0.001, which means the neighborhoods are statistically different for different k-values and the difference would be statistically as large or larger than

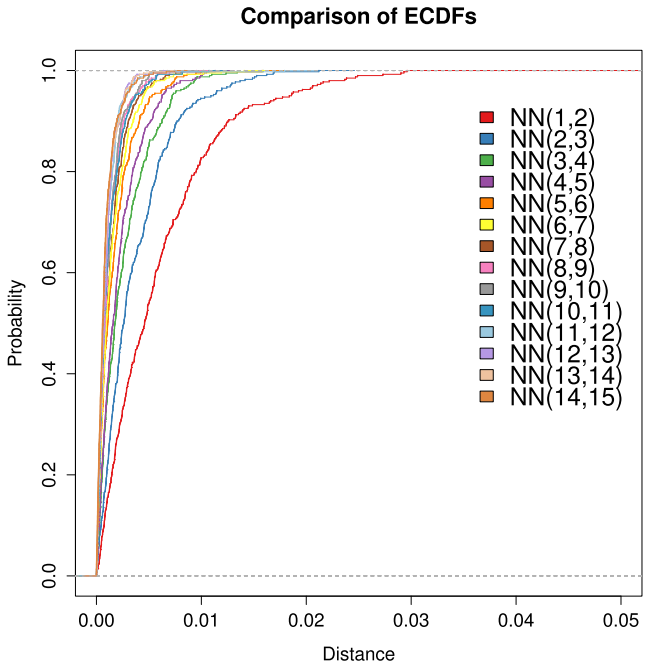

FIGURE 9. ECDFs of inter-neighbor distances using *DI-biased weights* for increasing k.

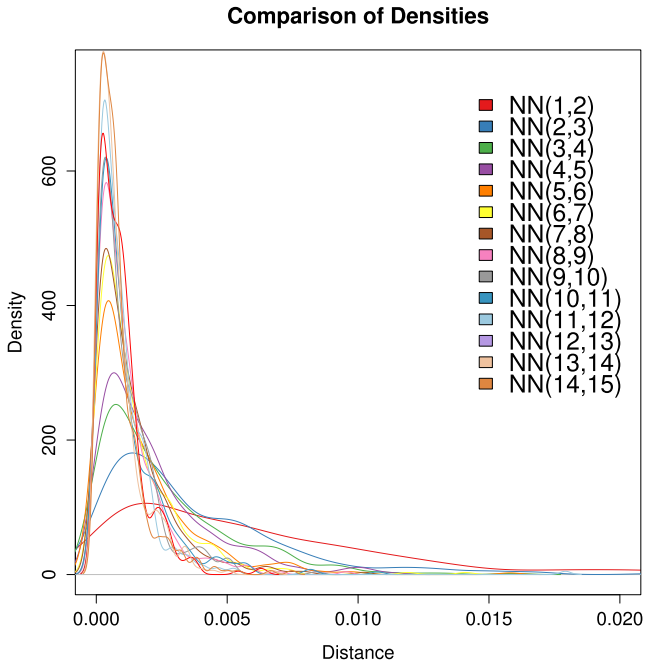

FIGURE 10. Density of inter-neighbor distances using *DI-biased weights* for increasing k.

the observed ones. If $d\_error(Median)$ and $d\_error(Mean)$ denote the improvement shown by the *DI-biased weights* method over the *baseline* method in terms of median and mean of prediction errors respectively for different k-values, the correlation between the corresponding D-values and the prediction errors, $d\_error(Median)$ and $d\_error(Mean)$ can be measured to show the relationship between the improvement in the dense quality neighborhood and the improvement of prediction accuracy. The last two columns of Table 7, show the Pearson's correlation scores of 0.484 (with median) and 0.393 (with mean). These positive correlation scores indicate that the improvement in terms of the dense quality neighborhood is correlated with the improvement in terms of prediction accuracy.

In total, the experimental results demonstrate that the heterogeneous contextual factors together with the driver intelligence can improve the quality of identified neighborhood significantly, which leads to a significant improvement in taxi queue wait time prediction.

VI. CONCLUSIONS

This paper focuses on the problem of taxi queue wait time prediction at the airport by using neighborhood based methods. Specifically, we investigate a large number of heterogeneous contextual factors associated with the JFK airport in New York City, including time of day, week, month, taxi trips to and from the airport, flight arrival times and passenger numbers as well as features related to weather conditions. To conduct this research, we generated a large-scale taxi queue wait time dataset for the research community by fusing three real-world datasets: taxi trip data, airport passenger arrival data and weather condition data. We first conducted a comprehensive analysis on the contextual features and the taxi queue wait times. Then, we proposed method to select relevant features and introduce a driver intelligence-biased feature weighting scheme to identify dense quality neighbors for k-NN based regression methods.

The experimental results show that the proposed driver intelligence-biased feature weighting scheme can enhance the performance of the state-of-the-art k-NN model for taxi queue time prediction. We show that the improvement in obtained results is statistically significant. Furthermore, the results obtained from inter-neighbor distance analysis demonstrate that the proposed method is able to identify dense neighborhoods for varying neighborhood size, which is also the reason for this significant improvement in prediction accuracy. Our research shows that such results obtained for queue wait time prediction can help manage airport queues related to taxi and passenger.

The results are restricted to prediction and do not offer optimum decision making solutions, which would require other different types of modelling. In future, we plan to include more contextual data sources such as local events and traffic congestion data with our taxi queue wait time dataset. Future research could provide optimal decision making for queue management by considering the estimated queue wait time. This will motivate the taxi drivers for an airport passenger pickup just after a precedent airport passenger drop-off to reduce their queue wait times at the airport taxi rank. Our generalized approach can be applied to other airports as the data becomes available.

REFERENCES

[1] M. A. Yazici, C. Kamga, and A. Singhal, "A big data driven model for taxi drivers' airport pick-up decisions in New York City," in *Proc. IEEE Int. Conf. Big Data*, Oct. 2013, pp. 37–44.

[2] R. Davis, T. Rogers, and Y. Huang, "A survey of recent developments in queue wait time forecasting methods," in *Proc. Int. Conf. Found. Comput. Sci.*, 2016, pp. 84–90.
[3] Y. Zhang, L. T. Nguyen, and J. Zhang, "Wait time prediction: How to avoid waiting in lines?" in *Proc. Ubicomp Adjunct*, 2013, pp. 481–490.
[4] M. F. Bulut, M. Demirbas, and H. Ferhatosmanoglu, "LineKing: Coffee shop wait-time monitoring using smartphones," *IEEE Trans. Mobile Comput.*, vol. 14, no. 10, pp. 2045–2058, Oct. 2015.
[5] K. Buza, A. Nanopoulos, and G. Nagy, "Nearest neighbor regression in the presence of bad hubs," *Knowl.-Based Syst.*, vol. 86, pp. 250–260, Sep. 2015.
[6] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *ACM Trans. Sensor Netw.*, vol. 46, no. 3, pp. 175–185, 1992.
[7] Y. Ren, G. Li, J. Zhang, and W. Zhou, "The efficient imputation method for neighborhood-based collaborative filtering," in *Proc. ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2012, pp. 684–693.
[8] J. Hilkevitch. (Jul. 12, 2015). *O'Hare Taxi Passengers, Drivers often in Holding Pattern, Chicago Tribune*. [Online]. Available: http://www.chicagotribune.com/business/chi-taxicabs-ohare-getting-around-met-20150316-column.html
[9] A. Anwar, M. Volkov, and D. Rus, "ChangiNOW: A mobile application for efficient taxi allocation at airports," in *Proc. 16th IEEE Int. Conf. Intell. Transp. Syst.*, Oct. 2013, pp. 694–701.
[10] C. Kamga, A. Conway, A. Singhal, and A. Yazici, "Using advanced technologies to manage airport taxicab operations," *J. Urban Technol.*, vol. 19, no. 4, pp. 23–43, 2012.
[11] B. Li *et al.*, "Hunting or waiting? Discovering passenger-finding strategies from a large-scale real-world taxi dataset," in *Proc. Percom Workshops*, 2011, pp. 63–68.
[12] A. Conway, C. Kamga, A. Yazici, and A. Singhal, "Challenges in managing centralized taxi dispatching at high-volume airports: Case study of John F. Kennedy international airport, New York City," *Transp. Res. Rec.*, vol. 2300, pp. 83–90, Nov. 2012.
[13] R. C. P. Wong, W. Y. Szeto, and S. C. Wong, "A cell-based logit-opportunity taxi customer-search model," *Transp. Res. C, Emerg. Technol.*, vol. 48, pp. 84–96, Nov. 2014.
[14] F. K. Putri and J. Kwon, "A distributed system for finding high profit areas over big taxi trip data with MognoDB and spark," in *Proc. IEEE Int. Congr. Big Data (BigData Congr.)*, Jun. 2017, pp. 533–536.
[15] W. Y. Szeto, R. C. P. Wong, S. C. Wong, and H. Yang, "A time-dependent logit-based taxi customer-search model," *Int. J. Urban Sci.*, vol. 17, no. 2, pp. 184–198, 2013.
[16] J. Tang, H. Jiang, Z. Li, M. Li, F. Liu, and Y. Wang, "A two-layer model for taxi customer searching behaviors using GPS trajectory data," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 11, pp. 3318–3324, Nov. 2016.
[17] L. Li, S. Wang, and F.-W. Wang, "An analysis of taxi driver's route choice behavior using the trace records," *IEEE Trans. Comput. Social Syst.*, vol. 5, no. 2, pp. 576–582, Jun. 2018.
[18] J. Guan, W. Wang, W. Li, and S. Zhou, "A unified framework for predicting KPIs of on-demand transport services," *IEEE Access*, vol. 6, pp. 32005–32014, 2018.
[19] C. Qing and W. Hao, "A methodology for measuring and monitoring congested corridors: Applications in manhattan using taxi GPS data," *J. Urban Technol.*, vol. 25, no. 4, pp. 59–75, 2018.
[20] B. Leng, H. Du, J. Wang, Z. Xiong, and L. Li, "Analysis of taxi drivers' behaviors within a battle between two taxi apps," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 296–300, Jan. 2015.
[21] H. Chen, B. Guo, Z. Yu, A. Chin, and C. Chen, "Optimizing taxiing with detours by using traffic dynamics and driving habits," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput., Adjunct (UbiComp)*, 2016, pp. 29–32.
[22] J. Lee, I. Shin, and G.-L. Park, "Analysis of the passenger pick-up pattern for taxi location recommendation," in *Proc. 4th Int. Conf. Netw. Comput. Adv. Inf. Manage.*, 2008, pp. 199–204.
[23] M. A. Hoque, X. Hong, and B. Dixon, "Analysis of mobility patterns for urban taxi cabs," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, 2012, pp. 756–760.
[24] L. Moreira-Matias, J. Gama, M. Ferreira, J. Mendes-Moreira, and L. Damas, "Predicting taxi–passenger demand using streaming data," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1393–1402, Sep. 2013.
[25] M. S. Rahaman, M. Hamilton, and F. D. Salim, "Predicting imbalanced taxi and passenger queue contexts in airport," in *Proc. Pacific Asia Conf. Inf. Syst. (PACIS)*, 2017, pp. 1–13.
[26] M. S. Rahaman, M. Hamilton, and F. D. Salim, "Queue context prediction using taxi driver knowledge," in *Proc. Knowl. Capture Conf.*, 2017, Art. no. 35.
[27] Y.-W. Lin and Y.-B. Lin, "Mobile ticket dispenser system with waiting time prediction," *IEEE Trans. Veh. Technol.*, vol. 64, no. 8, pp. 3689–3696, Aug. 2015.
[28] O. S. Pianykh and D. I. Rosenthal, "Can we predict patient wait time?" *J. Amer. College Radiol.*, vol. 12, no. 10, pp. 1058–1066, 2015.
[29] J. Goncalves, H. Kukka, I. Sánchez, and V. Kostakos, "Crowdsourcing queue estimations *in situ*," in *Proc. 19th ACM Conf. Comput.-Supported Cooperat. Work Social Comput.*, 2016, pp. 1040–1051.
[30] W. Wu, W. S. Ng, S. Krishnaswamy, and A. Sinha, "To taxi or not to taxi?—Enabling personalised and real-time transportation decisions for mobile users," in *Proc. IEEE Int. Conf. Mobile Data Manage.*, Jul. 2012, pp. 320–323.
[31] S. A. Dudani, "The distance-weighted k-nearest-neighbor rule," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-4, no. 1, pp. 325–327, Apr. 1976.
[32] A. T. Lora, J. M. R. Santos, A. G. Expósito, J. L. M. Ramos, and J. C. R. Santos, "Electricity market price forecasting based on weighted nearest neighbors techniques," *IEEE Trans. Power Syst.*, vol. 22, no. 3, pp. 1294–1301, Aug. 2007.
[33] M. Dialameh and M. Z. Jahromi, "A general feature-weighting function for classification problems," *Expert Syst. Appl.*, vol. 72, pp. 177–188, Apr. 2017.
[34] A. Wang, N. An, G. Chen, L. Li, and G. Alterovitz, "Accelerating wrapper-based feature selection with K-nearest-neighbor," *Knowl.-Based Syst.*, vol. 83, pp. 81–91, Jul. 2015.
[35] M. R. Al Iqbal, M. S. Rahaman, and S. I. Nabil, "Construction of decision trees by using feature importance value for improved learning performance," in *Proc. Int. Conf. Neural Inf. Process.*, 2012, pp. 242–249.
[36] M. Al Iqbal, M. Rahman, S. I. Nabil, and I. Ul Amin Chowdhury, "Knowledge based decision tree construction with feature importance domain knowledge," in *Proc. 7th Int. Conf. Elect. Comput. Eng. (ICECE)*, 2012, pp. 659–662.
[37] D. P. Vivencio, E. R. Hruschka, Jr., M. do Carmo Nicoletti, E. B. dos Santos, and S. D. C. Galvio, "Feature-weighted k-nearest neighbor classifier," in *Proc. IEEE Symp. Found. Comput. Intell.*, Apr. 2007, pp. 481–486.
[38] N. Jankowski and K. Usowicz, "Analysis of feature weighting methods based on feature ranking methods for classification," in *Proc. Int. Conf. Neural Inf. Process.*, 2011, pp. 238–247.
[39] P. J. García-Laencina, J. Sancho-Gómez, A. R. Figueiras-Vidal, and M. Verleysen, "K nearest neighbours with mutual information for simultaneous classification and missing data imputation," *Neurocomputing*, vol. 72, nos. 7–9, pp. 1483–1493, 2009.
[40] Z. Pan, Y. Wang, and W. Ku, "A new general nearest neighbor classification based on the mutual neighborhood information," *Knowl.-Based Syst.*, vol. 121, pp. 142–152, Apr. 2017.
[41] A. P. Sinha and H. Zhao, "Incorporating domain knowledge into data mining classifiers: An application in indirect lending," *Decision Support Syst.*, vol. 46, no. 1, pp. 287–299, 2008.
[42] D. W. Aha, "Feature weighting for lazy learning algorithms," *Feature Extraction, Construction and Selection*. Boston, MA, USA: Springer, 1998, pp. 13–32.
[43] N. J. Yuan, Y. Zheng, L. Zhang, and X. Xie, "T-finder: A recommender system for finding passengers and vacant taxis," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 10, pp. 2390–2403, Oct. 2013.

MOHAMMAD SAIEDUR RAHAMAN is currently pursuing the Ph.D. degree in computer science and IT with the School of Science, RMIT University, Australia. His research interests include human mobility and behavior analytics, spatio-temporal data mining, and computational intelligence.

YONGLI REN received the Ph.D. degree in information technology from Deakin University, Australia. He is currently a Lecturer of computer science and IT with the School of Science, RMIT University, Australia. His research interests include recommender systems, log analysis, user profiling, and data mining. He received the Alfred Deakin Medal for Doctoral Thesis in 2013 from Deakin University and the Best Paper Award from the IEEE/ACM ASONAM 2012 Conference.

MARGARET HAMILTON is currently an Associate Professor of computer science and IT with the School of Science, RMIT University, Australia, where she researches in the fields of human–computer interaction, mobility data mining, and computer science education. She is particularly interested in how technology can serve communities better, bringing experience in the design of surveys and statistical analyzes of qualitative and quantitative data, and interpretation of underlying trends in large datasets. Her work has involved funding from the Australian Research Council and Siemens. She has collaborated with Mornington Peninsula Shire over several years, first to research and deliver a ride-sharing app and, more recently, on developing socio-technical pedagogical approaches to designing apps for cities' wicked problems, such as public transport in Melbourne for elderly and disabled passengers and in particular, the Yarra Trams project for integrating the automatic passenger counting system into their tram transport management system.

FLORA D. SALIM received the Ph.D. degree from Monash University in 2009. She is currently a Senior Lecturer of computer science and IT with the School of Science, RMIT University. Her research interests are human mobility and behavior analytics, context and activity recognition, and urban intelligence. Her research in spatio-temporal data analytics, context recognition and behavior recognition, and prediction from sensor data has been evaluated across multiple projects, such as indoor monitoring and analytics in university and retail environments, driving behavior recognition, road risk analysis, and passenger movement analysis in airports. She is an Editorial Board Member of the *Pervasive and Mobile Computing* journal, an Expert Member of the International Energy Agency's Energy in Buildings and Communities programme (Annex 79), and a Panel Member of *JPI Urban Europe*. She received the Australian Research Council Postdoctoral Fellowship Industry from 2012 to 2015. She was a recipient of the RMIT Vice-Chancellor's Award for Research Excellence - Early Career Researcher 2016 and the RMIT Award for Research Impact - Technology 2018. She was a recipient of the Victoria Fellowship 2018 from the Victorian Government. She is also a Technical Program CommitteeVice-Chair of the 2018 IEEE PerCom. She is a Regular Invited Reviewer for the *ACM Transactions On Internet Technology*, *Pervasive and Mobile Computing* (Elsevier), the IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS, the IEEE TRANSACTIONS ON SERVICES COMPUTING, the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, the IEEE TRANSACTIONS ON CLOUD COMPUTING, and the *Data Mining and Knowledge Discovery* journal.