IJTE, Vol. 10, No. 2, Autumn 2022, pp. 999-1013
Research Paper

Predicting Dynamic Origin-Destination Matrix by Time Series Pattern Recognition

Seyed Omid Hasanpour Jesri[1], Mohsen Akbarpour Shirazi[2,*]

Received: 2021/05/24 *Accepted: 2021/09/08*

Abstract

Dynamic Origin Destination (OD) matrix estimation is a classic problem that has long been a subject of scholarly investigation. OD estimation is an essential prerequisite for transportation planning and traffic management. Despite the plethora of research on this subject, most models available in the literature fail to present the elegant characteristics of OD time series data. The patterns in OD time series break down into regular and particular patterns. However, most studies in literature focused on the regular type. Broadly the regular patterns are used to represent the general distribution patterns in time-dependent OD demands. Although, uncontrollable variables such as weather conditions, events, time, and crashes affect the OD patterns considerably. So, considering the impact of these uncontrolled variables, we developed a time series prediction algorithm model that can show both regular and particular patterns. The proposed model classifies historical data and estimates the class of coming demand. The clustering and association rule techniques are used in the proposed model to predict the coming OD. The bike riding data in Chicago was used to test the algorithm and the results suggest that the model can predict the class of OD with above 80% accuracy with a reasonable number of classes.

Keywords: Time Series, Origin Destination Matrix, Classification, Pattern Recognition, Traffic Management

* Corresponding author. E-mail: akbarpour@aut.ac.ir
[1] Ph.D. Candidate, Department of Industrial Engineering & Management Systems, Amirkabir University of Technology, Tehran, Iran
[2] Associate Professor, Department of Industrial Engineering & Management Systems, Amirkabir University of Technology, Tehran, Iran

1. Introduction

Extracting Origin-Destination (OD) demands is a vital prerequisite for planning and controlling transportation systems. OD demand Estimation represents the number of travelers departing from an origin at a particular time interval heading for a destination. The prediction of coming demand can highly affect the outcome of decisions, especially in the case of traffic management. Generally, OD demands are time-varying and influenced by uncontrollable variables such as weather conditions, events, time, and crashes. Historical data could show how these variables impact on the OD. By extracting OD patterns, we can learn from similar patterns in similar conditions.

Based on our survey, most studies in the related literature only consider the regular pattern. Generally the patterns in OD time series are broken down into regular and particular patterns. The former are used to represent the general distribution patterns in time-dependent OD demands and do not specify the effects of unique events, weather or incidents. Nor do they expose the daily or hourly fluctuations caused by Uncontrollable variables. These variables such as weather conditions, events, time, and crashes are considered in particular patterns. Such patterns are not necessarily unchangeable and are likely to recur in future. So, in this paper, by considering the uncontrolled variables, we have developed a time series prediction algorithm model that can show both regular and particular patterns. Many Advanced Traffic Management Systems (ATMS) require an accurate time- dependent OD demand as an input. Conventional DODE models tend to represent the average traffic patterns and OD demand on a typical day. For example, for two days of October 9 and 10, 2018, which are not weekends, conventional models suggest an almost invariant OD prediction, while if either day happens to be rainy or an accident occurs in either day, different OD patterns would apply. Despite the plenty of data available in the literature in this regard, custom patterns such as special hourly or daily patterns still remain under-researched. In this paper, we have proposed a data-driven model that estimates dynamic OD demands based on matching similar patterns in historical data. In this model, we learn from the past. In other words, the situations are not unprecedented. Trolling and delving into historical data can help derive similar patterns which can be used to figure out the behavior of an OD demand in future.

For a network in any time interval, OD is represented by a matrix with each element representing the travel demand between two specific nodes and specific time. In this paper, each matrix is considered as an object and the proposed model clusters objects based on similarity. The next model tries to predict the coming cluster based on search in historical data. The step by step research chart is shown in figure 1.

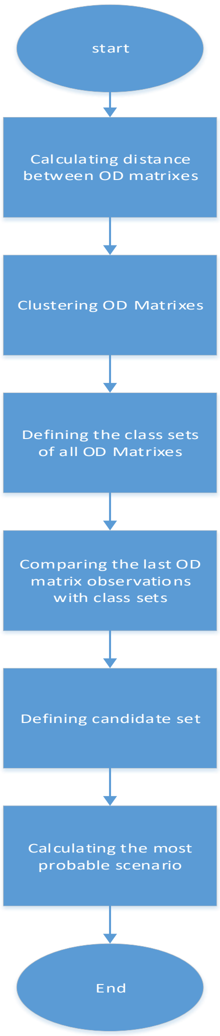

Figure 1. Research chart

Many methods have been proposed for the collection of OD demand data using traffic flow variables (e.g. speed data, count data, and link volume) by different traffic sensors. OD data collection methods can be generally categorized into two classes: fixed-sensor-based methods and trajectory-based methods. However, the accuracy of all of these methods is limited because of the immanent difficulty of tracking the origins and destinations of trips by detectors [Wenming et al., 2018]. With the advent of new communication technologies, not least the Internet of things (IoT), in the field of transportation, data gathering and processing does not prove as daunting and formidable a task as it used to be. Connected car systems enable the delivery of near real-time travel data without the driver having to take any action. The wireless communication technology enables data transfer between vehicles, infrastructure, and a consumer. There are two types of communication: vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I). V2V enables communication of one vehicle to other vehicles, while in the case of V2I, the data is exchanged between vehicles and the infrastructure. In a connected vehicle environment, each vehicle plays all the roles originally assigned to the sender, receiver, and router to transmit data to the vehicular network or ATMS. Presently the collected data is generally used for operational purposes, while it has great potential to support travel forecasting models, transportation system management, and traffic control. About 30 kinds of travel demand forecasting methods have been developed [Yang, et al., 2006]. However, many of them are purely theoretical and hardly applicable to real-world situations. Little wonder, then, exploring more practical travel demand forecasting has proven a perennial challenge in the travel demand forecasting field [Zhao et al. 2018].

With the development of big data, many new avenues have become open to us to estimate OD demand in ways never before imagined [Lam et al. 2021]. In this paper, we present a time series model which utilizes historical knowledge to predict future. It is supposed that accurate OD data has been retrieved so far and we intend to predict OD for near future. The clustering and association rule techniques are used in the proposed model to predict the coming OD. The presented model can show both regular and particular patterns.

Historical data shows trends which are useful in time series forecasting. Generally, the main component of time series is the secular trend (long-term direction); and the generalized least square model (GLS) is a commonly used model for OD trend estimation [Sherali and Park, 2001]. It is also important to note that secular trends cannot show particular patterns, and seasonal and cyclical movements in time series can only recognize periodic patterns. This is all the more reason for the necessity of developing a method capable of capturing and recognizing specific patterns. The proposed model aims to fill this gap in OD demand time series data. The main contributions of this paper can be summarized as follows:

- It proposes a data-driven model capable of recognizing and predicting particular patterns in OD time series data. It considers any OD data in each interval as an object and tries to find similar objects in time series data. It is a novel approach to OD prediction drawing on historical data.
- The model is dynamic and can relearn new patterns. In other words, the model detects a pattern out of events that have just happened patterns, which can be particular or general, while conventional models can detect general patterns after a long time.
- A numerical real-world experiment is done on bike riding data in Chicago for 4 straight years starting from 2013.The results point to the high accuracy of the model.

The remainder of the paper is organized as follows. The next section offers an overview of the existing relevant research in the literature. Sections 3 describes how the new Time series algorithm is used to forecast the future OD demand. Section 4 tests the proposed time series algorithm on bike riders in Chicago. Finally, Section 5 discusses the results and draws conclusions.

2. Literature Review

OD demand Estimation has been a subject of intense academic research for decades and, as a result, several DODE methods have been proposed [Karimi et al., 2020, Bell 1983; Hänseler, Molyneaux, and Bierlairea 2017; Kuusinen, Sorsa, and Siikonen 2015]. Almost all these methods aim to extract regular patterns, whereas little research has been done to recognize particular patterns, causing an egregious gap [Hasanpour et al., 2012].

Also analyzing traffic data has recently proven a highly popular research area [Ganin et al. 2019, Mohammadi et al., 2020, Torfehnejad and Jalali, 2018]. Zhang et al. (2008) surveyed the roles of data properties including count, speed and history OD data in the effectiveness of dynamic OD estimation. Data sources such as Bluetooth [Barceló et al., 2010], mobile phone [Calabrese et al., 2011; Iqbal et al., 2014, Bachir et al., 2019, Mohanty and puzdnukhov, 2020], Wi-Fi scanner [Hidayat et al.,2020], probe vehicles [Antoniou et al., 2006] automatic vehicle identification [Cao et al., 2021] and smart card-based automated fare collection [Zhen et al.,2018, Hamedmoghadam et al., 2021, Wu et al., 2021] were also utilized to estimate dynamic OD demands.

The most common model for OD demand estimation is the generalized least square model (GLS) [Sherali and Park, 2001]. Other modelling approaches include the maximum entropy model [Xie et al., 2011, Gonzalez et al., 2020, Lei et al., 2021], Bayesian theory [Hazelton, 2008; Castillo et al., 2008a, 2008b, 2014], Artificial neural network [zhang et al., 2021] and the state-space model [Zhou and Mahmassani, 2007; Alibabai and Mahmassani, 2009; Lu et al., 2015].

Wei and Zhen argue that the classic OD demand models are not qualified to identify particular patterns. Although OD demand is changing daily, it is somehow repetitive. The day-to-day patterns of OD demand cannot be verified by the current OD estimation methods. For this reason, demand patterns that evolve daily, weekly, monthly, seasonally and annually are not considered in such models; this is despite the sheer volume of data collected over the years [Wei et al., 2018].

To the best of the authors' knowledge, this paper is the first research on predicting OD based on a machine learning process. We present a novel data-driven model based on time series to predict short-term OD demand which is capable of identifying all particular patterns. Multivariate time series (MTS) prediction model can be applied to forecast future through estimating OD parameters data based on historical patterns. Machine learning algorithms has been increasingly used for time series forecasting [S. Makridakis et al., 2018], including artificial neural networks [G. Darbellay., 2000] and support vector machine [O. Kisi et al, 2015, Maldonado et al., 2019]. Although intelligent methods show higher performance over traditional statistical methods, however, these individual models are not able to obtain the optimal forecasting accuracy on account of their deficiencies. For example, ANNs are disposed to fall into a local optimum, as well as increasing unpredictable factors or computational aspects imposed these models [Zhang et al., 2018].

3. Estimating Dynamic OD Matrix by Time Series

The OD time series data has high potential in pattern extraction. In other words, these types of data are not random and are treated based on specific patterns. The basic assumption for forecasting is considering patterns in OD data, meaning the future OD demands are similar to those of the past. In that sense, by extracting historical patterns and matching recent data to the patterns, the model can forecast short-term horizons.

OD matrix for a network with $|N|$ node in period t is presented by A^t which is $|N| \times |N|$ matrix. A_{ij}^t is demand from node i to node j in t interval. The last observation is in T and all OD matrixes for all intervals make up the dataset. Any OD data in each interval is considered an object and the proposed model tries to find similar objects in time series data and predict the class of future OD for the forecast horizon. Time intervals can vary from a few minutes to several hours. For forecasting OD demands based on historical data, a heuristic algorithm is proposed involving 3 main steps: 1) clustering historical data and labeling members of any cluster; 2) comparing and adapting the class of λ^{th} last observed OD objects with historical data; and 3) analyzing the candidates and finding the most probable scenario for the defined horizon. The pseudocode of meta algorithm is shown in table 2. The output of the algorithm defines the class of the OD, meaning it does not estimate the OD values directly, while it predicts the class of OD which is close to OD values. The notation is presented in table 1.

Table 1. list of notations in this paper

Sets	
D	Dataset of OD matrices in all times
Class set	the set of OD classes labels
Last observed	the set of λ last observed class
Candidate set	the set of subsequences which are candidates for forecasting
Subsequence(t)	a subset of class set started from t
Candidate horizon(z)	the time of starting a candidate subsequence
Class	the set of OD matrices belong to same class
Parameters	
N	the size of the network
λ	number of last observed data for comparison
ρ	similarity threshold for acceptance
forecast horizon	length of time into the future for forecast
T	last time interval
δ	threshold of clustering
Variables	
D(N,N,t)	the OD demand matrix in time interval t
forecast	the forecasted class(es) based on highest score candidate
p_i	similarity between subsequence(i) and last observed data
$Score(c(i))$	final score of candidate(i)
Next sequences(t)	the label of class set(t+λ)

Table 2. Algorithm of prediction by time series

	Finding the best scenario (data set D, λ , ρ , forecast horizon)
1	Size(D,3) $\leftarrow$T
2	Class set $\leftarrow$cluster data(dataset D)
3	Last observed $\leftarrow \varnothing$
4	For i= λ:1
5	Last observed $\leftarrow$last observed $\bigcup$ class set(T-i+1)
6	End For
7	Candidate set $\leftarrow \varnothing$
8	For t=1:T-λ-1
9	Subsequence(t) $\leftarrow \varnothing$
10	For r=1:λ
11	subsequence(t) $\leftarrow$ subsequence(t) $\bigcup$ class set(t+r-1)
12	End For
13	End For
14	z$\leftarrow$1
15	For t=1:T-λ-1
	Finding the best scenario (data set D, λ , ρ , forecast horizon)
16	p$\leftarrow$calculate similarity(subsequence(t),last observed)
17	If p> ρ
18	candidate set$\leftarrow$candidate set $\bigcup$ subsequence(t)
19	Candidate horizon(z) $\leftarrow$t
20	Candidate sim(z) $\leftarrow$p
21	z$\leftarrow$z+1
22	End If
23	End For
24	S$\leftarrow$ size(candidate set)
25	rank candidate$\leftarrow$ranking(candidates set)
26	For v=1: $\lvert S \rvert$
27	Next sequences(v)$\leftarrow$check sequence(rank candidate(v),forecast horizon)
28	End For
29	forecast$\leftarrow$highest score frequency(Next sequences(v))

3.1. Clustering Historical Data

In this step, similar OD objects are clustered; in other words, the OD matrixes in the same class are close to each other and OD Matrixes in different classes are not very similar. Here we define the maximum intra-distance parameter with δ; it means distances between members of a specific cluster should be less than δ; consequently, as δ decrease, the number of clusters increases. The last observation of OD is placed in the first class and the next objects are compared with this. If the maximum intra-distance between the last observation and the intended object is less than δ, the object is added to this class; otherwise, a new class is created. This action repeats till all objects have a definite class.

Determining the number of clusters is a big issue in this problem. In fact, δ represents the level of the aggregation of data. For higher performance, the number of clusters should be well bounded. To eliminate the scale error, the matrixes are normalized with a standard score method called mapstd in the related pseudocode. The center of each cluster is also used calculate the distance. The pseudocode of the clustering approach is presented in Table 3.

Table 3. Clustering based on distance parameter

	cluster data(dataset D, δ)
1	(N,N,T) ⟵size(dataset D)
2	J⟵1
3	Class set⟵J
4	Class(J) ⟵D(N,N,T)
5	For t=(T-1):1
6	For j=1:J
7	Dist(t,j)⟵ dist(mapstd(D(N,N,t)),mapstd(Center(class(j))))
8	End For
9	Min dist ⟵min(dist(t,j))
10	If min dist<δ
11	s ⟵argmin(dist(j))
12	Class(s) ⟵class(s) $\cup$ D(N,N,t)
13	Class set(t)⟵s
14	Else
15	J⟵j+1
16	Class(J) ⟵D(N,N,t)
17	Class set(t)⟵J
18	End IF
19	End For

Table 4. center of class

	Center(A(I,R,K))
1	(I,R,K) ⟵size(class(j))
2	sum(i,r) ⟵0
3	For i=1:I
4	For r=1:R
5	For k=1:K
6	sum(i,r) ⟵sum(i,r)+class(i,r,k)
7	End For
8	End For
9	End For
10	For i=1:I
11	For r=1:R
12	Ave(i,r) ⟵sum(i,r)/K
13	End For
14	End For

Table 5. Distance between classes

	dist(D1(I,R),D2(I,R))
1	Power dist⟵0
2	For i=1:I
3	For r=1:R
4	Power dist⟵power((D1(I,R)-D2(I,R))+ Power dist
5	End For
6	End For
7	dist⟵sqrt(Power dist)

3.2. Comparing and Adapting

After clustering, the members of each cluster are labeled and any OD matrixes are replaced by the labels of its class which called class set. The λ^{th} last observed class is selected. We believe time series OD data has specified patterns and most of the time similar trends can be found in historical data.

For example, assume there are 7 clusters and the 5th last observed classes are [c1-c4-c3-c4-c5], the model searches the historical data for the exact or similar matches to last observed classes. The ρ parameter is defined as the similarity threshold for acceptance. If the similarity between any subsequence of historical data and last observation is greater than the similarity threshold, then the subsequence is considered a candidate and all the candidates are entered to a set to be evaluated. The pseudocode of comparison and adapting approach is presented in Table 6.

The number of members in the candidate set determines the type of patterns. The few candidates represent a particular pattern is extracted. To other words these patterns are not occurred before frequently and most of the models in literature are not capable to recognize this type of patterns. The regular patterns have been occurred repeatedly so there are more number of candidates.

Table 6. Calculation of similarity

	Calculate similarity(subsequence(t),last observed)
1	For d =1: λ
2	If subsequence(t+d-1) = last observed(d)
3	Similarity(t) ← Similarity(t)+1
4	Else
5	dist ← dist(N(t+d-1), N(T-λ+d))
6	Distance(t) ← Distance(t)+dist
7	End IF
8	p ← similarity(t)/ λ
9	End For

3.3. Finding the Most Probable Scenario

Assume the number of candidates in the candidate set is n and forecast horizon is equal to m. Based on the literature, two factors contribute to the prediction power. The first one is what percentage of classes match exactly. In other words, the higher the similarity between any candidate and the last observed data, the more probable it is to occur [Argaval et al., 1993]. For example, if 4 of 5 classes match, the percentage of similarity is 80%. The formula of calculating the similarity between classes of a candidate and the last observed data is presented in equation (1).

The second factor is the time the candidates occur; the time difference between the candidates and the last observed data shows which patterns have been done recently and if the time of a candidate occurring is closer to the current time, it could be more reliable. In light of both these factors, the final score of each candidate is calculated based on equation (2), where α is a constant less than 1, Δt is the time interval between candidate i and the last observed data and p_i is the percentage of the exact match. The phrase $\alpha(1-\alpha)^{\Delta(T-\lambda-t_i)}$ is adapted from the exponential smoothing function, while the model explicitly uses an exponentially decreasing score for past observations [Gardner, 2006].

$$p_i = \frac{\sum_{d=1}^{\lambda}(1 - |sgn(subsequence(t_i + d - 1) - lastobserved(d))|)}{\lambda}$$

$$Score\big(c(i)\big) = \alpha(1-\alpha)^{\Delta(T-\lambda-t_i)}(p_i)$$

Forecasting is done based on the sequence of classes in the candidate with the highest score. The model can be improved by learning ability; once the subsequent periods have occurred and once it has been ascertained which classes have happened, the parameters of the model can be fine-tuned by learning techniques such as neural network. The pseudocode of ranking and scoring is presented in

Table *7*, Table 8 and Table 9.

Table 7. Ranking of the candidate set

Ranking
For s=1: $\lvert S \rvert$
For h=1: $\lvert S \rvert$
If p(candidate set(s)) >p((candidate set(h))
Rank(s)=rank(s)+1
Else IF p(candidate set(s)) =p((candidate set(h)) & Distance(candidate set(s)) <= Distance((candidate set(h))
Rank(s)=rank(s)+1
End For
End For
For s=1: $\lvert S \rvert$
Rank candidate(rank($\lvert S \rvert$ -s+1)) $\leftarrow$ candidate set(s)
Ranked horizon(rank($\lvert S \rvert$ -s+1)) $\leftarrow$ Candidate horizon(s)
Ranked sim(rank($\lvert S \rvert$ -s+1)) $\leftarrow$ candidate sim(s)
End For

Table 8. Check Sequence

Check sequence(v)(rank candidate(v),Forecast horizon)
Check sequence(v) $\leftarrow \varnothing$
For p=1: Forecast horizon
Check sequence(v) $\leftarrow$ Check sequence(v) $\cup$ class set(ranked horizon(v)+ λ +p-1)
End For

Table 9. Scoring of the candidate set

Highest score frequency(α, candidate set)
For v=1: $\lvert S \rvert$
score(v) $\leftarrow$ α * power(1-α, T-$\lambda - candidate\ horizon(v)) * candidate\ sim(v)$
End For
Forecast $\leftarrow$ next sequence(argmax(score))

4. Numerical Experiments

The data for test has to have two properties. Firstly, it has to have the Origin and Destination information by time and, secondly, it has to be available for a long period of time. Based on these requirements historical trip data for bike riders in Chicago was chosen to test the proposed time series algorithm which is open source and free and also contains long term OD data. The dataset has 12 fields including: 1(trip id)- 2(start time)- 3(stop time)- 4(bike id)- 5(trip duration)- 6(from-station-id)- 7(from-station-name)-8(to-station-id)- 9(to_station_name)-10(user type)- 11(gender)- and 12(birth year). For four straight years starting from 2013, 10% of all stations have been studied. Time intervals are considered in several scenarios: 8hour, 12hour, and 24hour. The OD matrix is square of order 30 and number of time interval is 4386, 2924 and 1462 for mentioned scenarios.

The data set was split into two subsets: training set and validation set.70% of dataset is used for training and 30% is used for validation of Model [S. Maldonado et al., 2019].
Threshold represents the similarity between the OD matrixes in different intervals figure 2 shows the relation between the threshold and the number of clusters under 4 scenarios: 8hour interval, 12 hour interval, 24hour interval and random data.

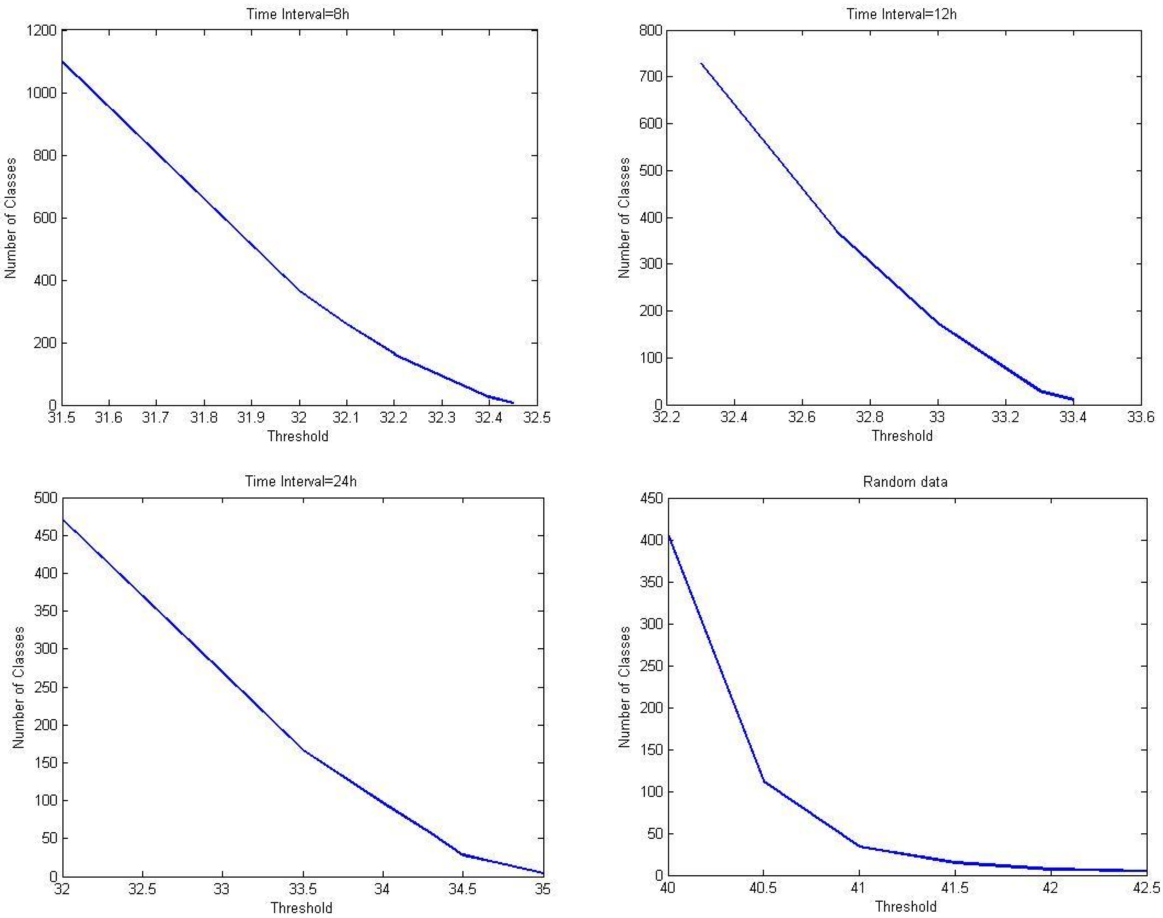

Figure 2. The relation between Threshold and number of classes

When the time interval is longer, the similarity level drops and the threshold rises. For random data, threshold is at least %25 greater than time series data for a specific number of classes, which indicates the similarity between random data is much less than the similarity between time series data. There is an adverse correlation between true prediction rate and the number of classes such that as the number of classes increases, the true prediction rate decreases. The sensitivity of the true prediction rate to the number of classes is higher when the time intervals are longer. In random data, the true prediction rate for all numbers of classes is very low and close to zero, showing that there is no pattern in random data to be used for prediction figure 3 illustrates the true prediction rate based on the number of classes for forecast_horizon=1.

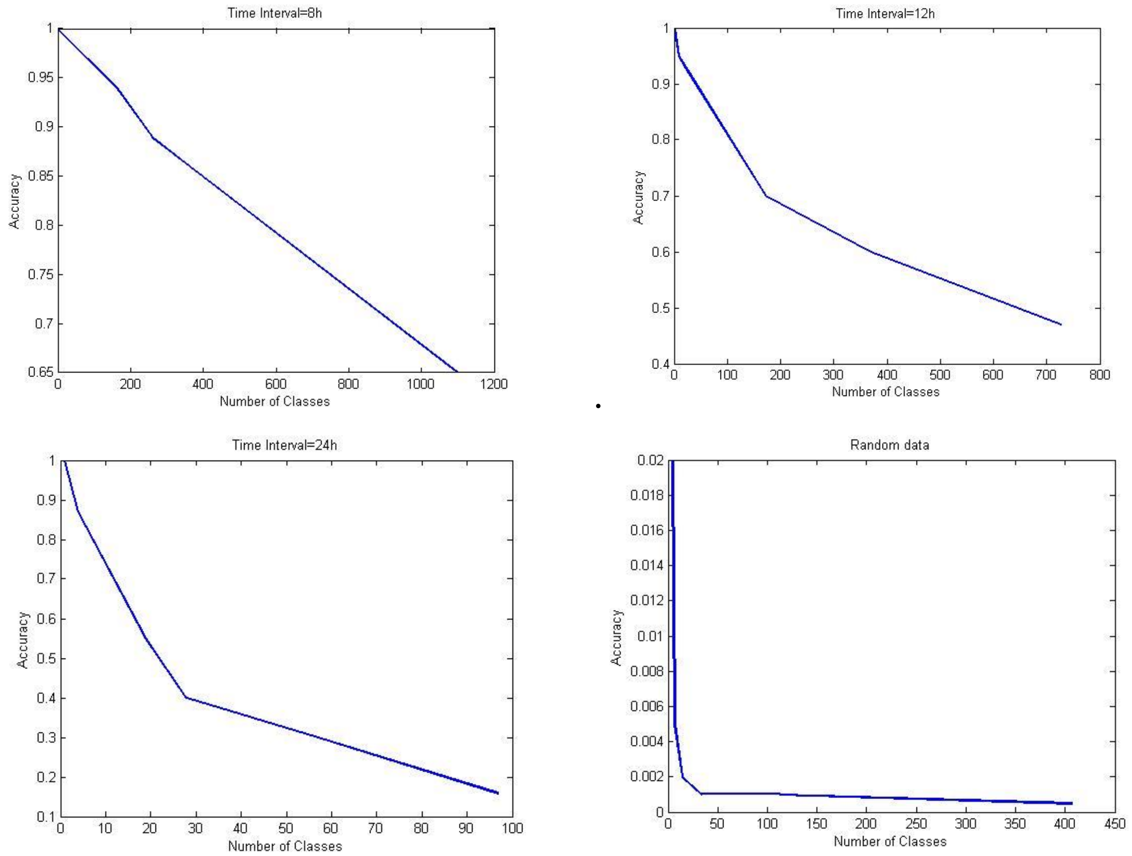

Figure 3. True prediction rate for different number of classes

4.1. Validation of Model

After the model is fitted well with training data set, the model is validated by validation data set to assess the accuracy of the fitted model. For testing the model, the random integer point is generated and considered as the last observation in the presented model and the algorithm predicts the next label of the class. For this purpose, first the actual classes of validation data set are defined, next the model predicts the class of validation data set and finally the predicted classes are compared to actual class in validation data set. The percentage of true predictions defines the accuracy of the algorithm.

The parameters of the model and the accuracy for different test scenarios are presented in Table 10. It is much expected that, accuracy decreases when forecast horizons increase also while the time intervals are shorter, the number of time data records increases and the model searches in larger space to find similarity so accuracy in predicting increases.

Table 10. Prediction rate for different Scenarios

Test scenario	λ	Number of classes	Time Interval(hour)	Prediction rate for 1st next class
1	12	4	8	89.3 %
2	12	8	8	75.1%
3	6	4	8	80.6%
4	6	8	8	68.8%
5	12	4	24	86.2%
6	12	8	24	72.7%
7	6	4	24	74.0%
8	6	8	24	61.6%

Error! Reference source not found. represents the accuracy of True prediction for validation data set for different forecast horizons for the first 1st scenario of table10.

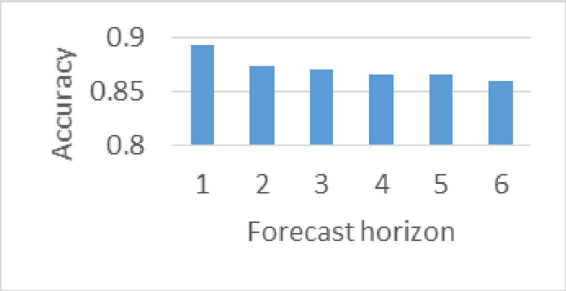

Figure 4. The relation between the forecast horizon and accuracy

The average distance between exact OD matrix and true forecasted class's values is shown in **Error! Reference source not found.**. The distance rate is almost excusable by considering stochastic fluctuation.

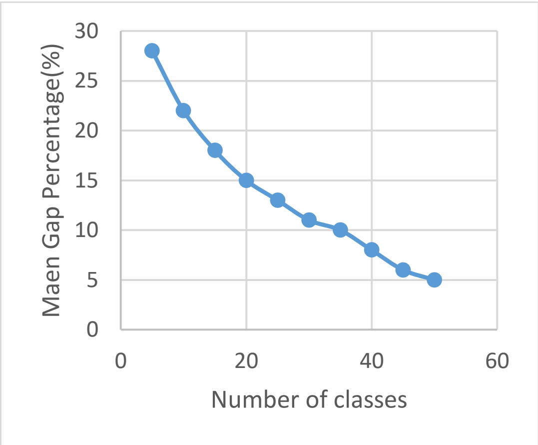

Figure 5. Mean gap between true forecasted class and exact OD values (%)

Many criteria are used to evaluate the forecasting performance of different methods in empirical studies. In this paper we employ MAPE (mean absolute percentage error) as the forecast accuracy to compare our proposed model with K-nearest neighbor (KNN) regression which is a very popular model in literature for forecasting traffic time series [Yin and Shang., 2016]. The result is shown in Table 11. The results prove that the proposed model performs better than the KNN in all time intervals scenarios.

Table 11. Comparison of MAPE value

Time Interval	8 Hour	12 Hour	24 Hour
Proposed Model	4.84	5.02	5.55
KNN	5.63	5.91	6.17

5. Conclusion

With the advent of innovative communication technologies in the field of transportation, new possibilities have become open to us as far as traffic management and planning are concerned. Real-time data processing is one of these possibilities that can enable OD prediction. This paper proposes a data-driven time series algorithm drawing on historical patterns in demand data and predicts the future based on similarities between recent data and historical data. The idea is to find similarities between historical data and the last observed data. A clustering approach is used to find similar OD Objects, an algorithm is proposed to find similar patterns in historical data, and, finally, future OD demands are predicted based on the most scored scenario. The near real-time OD prediction can help traffic planners to manage traffic conditions by policy-making based on predicted OD. The results of the test on bike riding data in Chicago show an acceptable rate of prediction accuracy for a reasonable number of clusters. One interesting area for future studies can be providing a learning module for optimizing parameters of time series algorithm; the presented algorithm has parameters some of which can affect the accuracy of the model. In this paper, the parameters are set by trial and error, while artificial neural network or any learning system seem to be a better fit. The present authors intend to work on the issues raised here in near future.

6. Appendix A. Supplementary Material

Data associated with the time series part can be found on the following website: (https:// www.divvybikes.com/system-data)

7. References

– Alibabai, H., Mahmassani, H.S., (2009) "Dynamic origin-destination demand estimation using turning movement counts", Transport. Res. Rec.: J. Transport. Res. Board 2085 (1), 39–48.

– Antoniou, C., Ben-Akiva, M., Koutsopoulos, H.N., (2006) "Dynamic traffic demand prediction using conventional and emerging data sources" In: IEE Proce.-Intell. Transp. Syst., vol. 153. IET, pp. 97–104.

– Bachir, L., Khodabandelou, Gauthier, El Yacoubi, Puchinger, (2019), "Inferring dynamic origin-destination flows by transport mode using mobile phone data". Transportation Research Part C: Emerging Technologies, Volume 101, Pages 254-275.

– Barceló, J., Montero, L., Marqués, L., Carmona, C., (2010). "Travel time forecasting and dynamic origin-destination estimation for freeways based on bluetooth traffic monitoring" Transport. Res. Record: J. Transport. Res. Board (2175), 19–27.

– Calabrese, F., Di Lorenzo, G., Liu, L., Ratti, C., (2011). "Estimating origin-destination flows using mobile phone location data". IEEE Pervasive Computing 10 (4), 0036–44.

– Cao, Y., Tang, K., Sun, J., Ji, Y., (2021)."Day-to-day dynamic origin–destination flow estimation using connected vehicle trajectories and automatic vehicle identification data". Transportation Research Part C: Emerging Technologies Volume 129, 103241

– Castillo, E., Menéndez, J.M., Jimenez, P., (2008) b. "Trip matrix and path flow reconstruction and estimation based on plate scanning and link observations". Transport. Res. Part B: Methodol. 42, 455–481.

– Castillo, E., Menéndez, J.M., Sánchez-Cambronero, S., (2008) a. "Traffic estimation and optimal counting location without path enumeration using Bayesian networks." Computer-Aided Civil Infrastruct. Eng. 23 (3), 189–207.

– Castillo, E., Menéndez, J.M., Sánchez-Cambronero, S., (2014). "A hierarchical optimization problem: estimating traffic flow using gamma random variables in a Bayesian context". Comput. Operation Res. 41, 240–251.

– Ganin, A. A., Mersky, A. C., Jin, A. S., Kitsak, M., Keisler, J. M., & Linkov, I. (2019). "Resilience in intelligent transportation systems (ITS). Transportation", Research Part C: Emerging Technologies, 100, pp. 318-329.

– Gardner, E. S., (2006). "Exponential smoothing: The state of the art" — Part II. International Journal of Forecasting, 22(4), 637–666.

– Gonzalez-Calderon, A., JairoPosada-Henao, SusanaRestrepo, (2020). "Temporal origin–destination matrix estimation of passenger car trips. Case study: Medellin, Colombia". Case Studies on Transport Policy Volume 8, Issue 3, 1109-1115.

– Hamedmoghadama, H., Hai L. Vu, Jalili, M., Saberi, M., Stone, L., Hoogendoorn, S. (2021). "Automated extraction of origin-destination demand for public transportation from smartcard data with pattern recognition" Transportation Research Part C: Emerging Technologies Vol. 129, p. 103210.

– Hänseler, Flurin S., Nicholas A. Molyneaux, and Michel Bierlairea, (2017), "Estimation of Pedestrian Origin-Destination Demand in Train Stations", Transportation Science, 51.3 ,981–97.

– Hasanpour S.O, Ahmadi A, Karimi B, Akbarpour M. (2012). "Hierarchical Data clustering Model for Analyzing Passengers Trip in Highway". International Journal of Industrial Engineering & Production Research. 23(4), 253-259.

– Hazelton, M.L., (2008). "Statistical inference for time varying origin-destination matrices" Transport. Res. B: Methodol. 42 (6), 542–552.

– Hidayat, A., Tarebo, S., Yaginuma, H., (2020). "Bus Passenger Volume and Origin-Destination Based on Field Surveys Using a Wi-Fi Scanner" Transportation Research Procedia, Volume 48, Pages 1376-1389.

– Iqbal, M.S., Choudhury, C.F., Wang, P., González, M.C., (2014). "Development of origin-destination matrices using mobile phone call data". Transport. Res. C: Emerg. Technol. 40, 63–74.

– Karimi, H., Shetab Boushehri, Nasiri, (2020). "Origin-Destination Matrix Estimation Using SocioEconomic Information and Traffic Counts on Uncongested Networks", International Journal of Transportation Engineering, Vol. 8/ No.2/ (30)

– Kisi, O., Shiri, J., Karimi, S., Shamshirband, S., Motamedi, S., Petkovic, D., Hashimd, R., (2015) "A survey of water level fluctuation predicting in Urmia Lake using support vector machine with firefly algorithm", Appl. Math. Comput. 270 731–743.

– Kuusinen, Juha Matti, Janne Sorsa, and Marja Liisa Siikonen, (2015) "The Elevator Trip Origin-Destination Matrix Estimation Problem", Transportation Science, 49.3, 559–76.

– Lam, W., Shao, H., Cao, S., Yang, H., (2021). "Origin–Destination Demand Estimation Models". International Encyclopedia of Transportation 2021, Pages 515-518.

– Lee, J.-G., Han, J., Whang, K.-Y, (2007). "Trajectory clustering: a partition-and-group framework". In Proc. Of the 2007 ACM SIGMOD International Conference on Management of Data (SIGMOD'07).

– Lei, D., Chen, X., Cheng, L., zhang, L., Wang, P., Wang, K., (2021). "Minimum entropy rate-improved trip-chain method for origin–destination estimation using smart card data". Transportation Research Part C: Emerging Technologies Volume 130, 103307

– Lu, Z., Rao, W., Wu, Y., Guo, L., (2015). "A Kalman filter approach to dynamic OD flow estimation for urban road networks using multi-sensor data". J. Adv. Transport. 49 (2), 210–227.

– Makridakis, S., Spiliotis, E., Assimakopoulos, V., (2018),"Statistical and machine learning forecasting methods: Concerns and ways forward", PLoS One 13 (3) e0194889.

– Maldonado, Sebastián, Agustín González, and Sven Crone. (2019). "Automatic Time Series Analysis for Electric Load Forecasting via Support Vector Regression". Applied Soft Computing 83: 105616.

– Mohammadi, M., Dideban, A., Lesani, A., Moshiri, B. (2020). "An Implementation of the AI-based Traffic Flow Prediction in The Resilience Control Scheme." International Journal of Transportation Engineering, Vol. 8/ No.2, pp. 185-198.

– Mohanty, S., Puzdnukhov, A., (2020), “Dynamic Origin-Destination Demand Estimation from Link Counts, Cellular Data and Travel Time Data”. Transportation Research Procedia, Volume 48, 1722-1739.

– Sherali, H., Park, T., (2001). “Estimation of dynamic origin-destination trip tables for a general network”. Transport. Res. Part B: Methodol. 35 (3), 217–235.

– Torfehnejad, H., Jalali, A. (2018). "Traffic condition detection in freeway by using autocorrelation of density and flow”. International Journal of Transportation Engineering, Vol. 6, No. 1, pp 85-98.

– Wei M., Zhen (Sean) Q., 2018. “Estimating multi-year 24/7 origin-destination demand using high-granular multi-source traffic data.” Transportation Research Part C 96, 96–121.

– Wenming R, Yao-Jan W, Jingxin X, Jishun O, Robert K., (2018). “Origin-destination pattern estimation based on trajectory reconstruction using automatic license plate recognition data.” Transportation Research Part C 95, 29–46.

– Wu, L., Kang, J., Chung, Y., Nikolaev, A., (2021). “Inferring origin-Destination demand and user preferences in a multi-modal travel environment using automated fare collection data”. Omega, Volume 101, June 2021, 102260.

– Xie, C., Kockelman, K., Travis, W., (2011). “A maximum entropy-least squares estimator for elastic origin–destination trip matrix estimation” Transport. Res. Part B: Methodol. 45 (9), 1465–1482.

– Yang, Z., Wang, Y., & Guan, Q. (2006). “A short-term traffic flow forecasting method based on support vector machine”. Journal of Jilin University (Engineering Science Edition), 36(6), 881–884.

– Yi Yin, Pengjian Shang, (2016), “Forecasting traffic time series with multivariate predicting Method”, Applied Mathematics and Computation 291, 266–278.

– Zhang, H., Nie, Y., Qian, Z., (2008). “Estimating time-dependent freeway origin-destination demands with different data coverage: sensitivity analysis.” Transport. Res. Record: J. Transport. Res. Board (2047), 91–99.

– Zhang, J., Che. H., Chen. F., Ma, W., He, Z., (2021). “Short-term origin-destination demand prediction in urban rail transit systems: A channel-wise attentive split-convolutional neural network method”, Transportation Research Part C: Emerging Technologies Volume 124, 102928.

– Zhang, Xiaobo, and Jianzhou Wang. (2018). “A Novel Decomposition-ensemble Model for Forecasting Short-term Load-time Series with Multiple Seasonal Patterns”. Applied Soft Computing Journal 65: 478–94.

– Zhen Chen., Wei Fan., (2018). “Extracting bus transit boarding stop information using smart card transaction data”. J. Mod. Transport. 26(3), 209–219.

– Zhou, X., Mahmassani, H., (2007). “A structural state space model for real-time traffic Origin-Destination demand estimation and prediction in a day-to-day learning framework.” Transport. Res. Part B: Methodol. 41 (8), 823–840.