Network-wide Crowd Flow Prediction of Sydney Trains via customized Online Non-negative Matrix Factorization

Yongshun Gong
University of Technology, Sydney
Australia
yongshun.gong@student.uts.edu.au

Zhibin Li
University of Technology, Sydney
Australia
zhibin.li@student.uts.edu.au

Jian Zhang
University of Technology, Sydney
Australia
jian.zhang@uts.edu.au

Wei Liu
University of Technology, Sydney
Australia
wei.liu@uts.edu.au

Yu Zheng
JD Urban Computing Business Unit and JD Intelligent City Research
China
msyuzheng@outlook.com

Christina Kirsch
Sydney Trains-Operational Technology
Australia
Christina.Kirsch@transport.nsw.gov.au

ABSTRACT

Crowd Flow Prediction (CFP) is one major challenge in the intelligent transportation systems of the Sydney Trains Network. However, most advanced CFP methods only focus on entrance and exit flows at the major stations or a few subway lines, neglecting Crowd Flow Distribution (CFD) forecasting problem across the entire city network. CFD prediction plays an irreplaceable role in metro management as a tool that can help authorities plan route schedules and avoid congestion. In this paper, we propose three online non-negative matrix factorization (ONMF) models. ONMF-AO incorporates an Average Optimization strategy that adapts to stable passenger flows. ONMF-MR captures the Most Recent trends to achieve better performance when sudden changes in crowd flow occur. The Hybrid model, ONMF-H, integrates both ONMF-AO and ONMF-MR to exploit the strengths of each model in different scenarios and enhance the models' applicability to real-world situations. Given a series of CFD snapshots, both models learn the latent attributes of the train stations and, therefore, are able to capture transition patterns from one timestamp to the next by combining historic guidance. Intensive experiments on a large-scale, real-world dataset containing transactional data demonstrate the superiority of our ONMF models.

CCS CONCEPTS

• **Information systems → Spatial-temporal systems**;

KEYWORDS

Crowd flow prediction; online non-negative matrix factorization; city trains network

CIKM '18, October 22–26, 2018, Torino, Italy

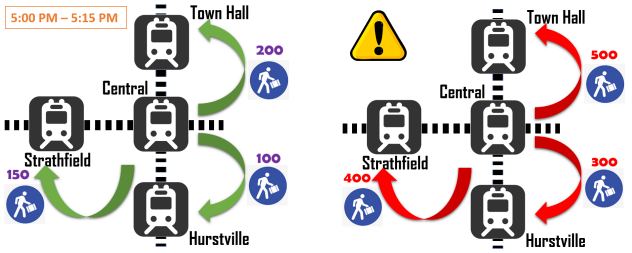
ACM ISBN 978-1-4503-6014-2/18/10…$15.00
https://doi.org/10.1145/3269206.3271757

ACM Reference Format:
Yongshun Gong, Zhibin Li, Jian Zhang, Wei Liu, Yu Zheng, and Christina Kirsch. 2018. Network-wide Crowd Flow Prediction of Sydney Trains via customized Online Non-negative Matrix Factorization. In *The 27th ACM International Conference on Information and Knowledge Management (CIKM '18), October 22–26, 2018, Torino, Italy*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3269206.3271757

1 INTRODUCTION

Predicting crowd flows in city trains network is strategically important for intelligent transportation systems because of the benefits it brings to many metro management and urban optimization services, such as congestion avoidance, route scheduling, public safety, and so forth [14, 30]. Thanks to the Opal card (Transportation NSW's smartcard ticketing system for travel on public transport, including Sydney Trains), a large amount of transactional data is now available that contains very detailed information. Based on these useful data, a number of applicable passenger flow prediction models have been proposed to enhance the metro services and improve operational performance of transit authorities [16, 29]. To date, most of these applications focus on capturing frequent passenger movement patterns or the entrance and exit flows for major stations.

However, in many real-world applications, concentrating solely on entrance and exit flows does not provide adequate information, managers also need to know potential passenger distributions, i.e., CFD forecasts. Figure 1 (a) illustrates an example CFD prediction. The model predicts that 450 passengers will "tap on" (enter) with their Opal tickets at Central Station between 5:00 PM and 5:15 PM and 200 will "tap off" (exit) at Town Hall, 150 at Strathfield, and 100 at Hurstville. A CFD model can illustrate the crowd flows among all these stations, which is significant for passenger route planning, train scheduling, and crowd warning systems. These models could be especially useful for predicting passenger flows during irregular events, such as train faults, emergencies, and public events, where passenger flows may suddenly surge over a short time span. With a strong CFD model, a transport administrator could forecast abnormal flow patterns and plan crowd evacuations for all affected stations to ensure public safety and/or maintain the normal train scheduled, as shown in Figure 1 (b).



(a) 450 passengers enter at Central between 5:00 to 5:15 pm, and the distribution of this entrance flows

(b) A congestion warning for all possibly affected stations when occurring non-recurrent events

Figure 1: An example of Crowd Flow Distribution

The existing techniques for addressing crowd flow prediction problems are mainly based on regression strategies like auto-regressive integrated moving averages (ARIMA) [24] or Gaussian processes (GP) [31]. Other strategies, such as neural networks [23], probability trees [11] and wavelet-SVM [20] have also been proposed as solutions to passenger flow prediction problems. However, to the best of our knowledge, none of these techniques are able to predict crowd flows across an entire train network. Such a task is much more difficult than simply forecasting entrance and exit passenger flows from each station as this type of problem contains three intrinsic challenges.

• *High computational complexity.* Network-wide CFD prediction needs to calculate *all* potential flows. Most existing methods only focus on entrance/exit flows at major stations or a few subway lines [4, 5, 14, 20, 23], which is already computationally expensive or requires a lengthy off-line training period that is difficult to adapt to network-wide forecasting.

• *Real-time data.* In a real-time system, there is a time lag between when a traveler enters a station and exits another. Hence, only partial data are available until the traveler completes their journey, making it impossible to get exact origin-destination records to reveal travel movements and durations during the present timespan.

• *Unpredictability.* It is difficult to detect the abnormal flows associated with many kinds of anomalies, such as emergencies, train faults, and traffic controls.

To address these challenges, we have developed several online non-negative matrix factorization models (ONMF) for network-wide crowd flow prediction. Our models are able to forecast both entrance/exit flows and CFD by combining the current trends in flow with historic guidance. ONMF has been recently used as an excellent strategy for successfully solving spatiotemporal network problems in traffic flow prediction [1, 7]. In these strategies, ONMF is used to handle network-wide problems in a similar way to the traditional NMF techniques applied in community and recommendation networks [28], but the model also incorporates temporal information to detect latent factors and track their evolution as the data evolves [3]. Based on the idea of network factorization, a CFD network is embedded into two latent spaces; one represents the attributes of the entrance stations, the other represents the exit stations. Our model, called ONMF-AO, goes a step further to capture the dynamic movements of these latent attributes from one timestamp to the next via the average optimization of transition matrices and historic guidance. ONMF-AO relies on an averaging strategy to describe the CFD trends in a given time window. However, CFDs will change dramatically when an abnormal incident occurs or during rush hour. And, in these situations, ONMF-AO is not always able to capture the sudden changes in flows by solely considering the averages of several previous CFD trends.

Motivated by this issue, we further developed another ONMF model based on the most recent CFD trends and historic guidance, called ONMF-MR, to handle irregular events. In this model, changes in CFD are primarily predicted by their most recent tendencies. ONMF-MR performs better than ONMF-AO when passenger flows suffer a sharp change. Additionally, given that each model demonstrates relatively better strengths in different scenarios, we propose a third hybrid model, called ONMF-H, that incorporates both ONMF-AO and ONMF-MR for use in real-world applications. The main contributions of this paper are summarized as follows:

• We formulate network-wide CFD prediction problems as a graph network problem and propose a data-driven forecasting model, called ONMF-AO, that combines current flow trends with historic guidance to address the three inherent challenges associated with network-wide crowd flow prediction.

• To further improve the effectiveness of the model in real-world situations, we designed another extended ONMF model, called ONMF-MR, that is able to adapt to sudden changes in crowd flows.

• A hybrid model, called ONMF-H, integrates both ONMF-AO and ONMF-MR to address a variety of challenging traffic scenarios.

• We compare each of the proposed models with five available prediction methods in a set of intensive experiments on a large, real-world Opal Card dataset covering Sydney Trains. The experiments assess the models' effectiveness from four perspectives, including: CFD predictions across the entire network at different timestamps, CFD predictions at major stations, comparisons between weekdays and weekends, and comparisons between peak and non-peak times. The experimental results show that ONMF-AO achieved the best results for the weekend tests, and ONMF-H proved to be more stable and effective for all weekday tests.

The rest of this paper is organized as follows. Section 2 discusses the related work. In Section 3, we formalize the problem of network-wide CFD prediction for Sydney Trains. The proposed CFD prediction model ONMF-AO is detailed in Section 4. Section 5 proposes ONMF-MR model and the hybrid strategy, ONMF-H, to improve the forecasting accuracy. The experimental results are presented in Section 6. Section 7 concludes the paper.

2 RELATED WORK

For the consideration of readers, current studies on crowd flow prediction are discussed first, followed by the common applications for ONMF.

2.1 Crowd Flow Prediction

Thanks to the increasingly wide availability of transactional data, there has been a recent explosion of studies on crowd flow prediction problems. Most are focused on entrance/exit flows at specific stations within a network or on a few select subway lines. For instance, Wei et al. proposed an approach for exploring the time variants in short-term passenger flows on a single metro line [5] and developed a modified forecasting method using neural networks [23]. Several scholars have studied prediction methods based on entrance/exit flows at a specific set of stations [4, 11, 14, 20].

Table 1: Symbol Description

Symbols	Descriptions
G; n	city trains network; number of stations
D_{OD}/D_{DO}	entrance or exit CFD matrices
W; H	latent space matrices
k	the number of dimensions of latent attributes
T	the number of consecutive snapshots in a window
A; B	transition matrices of W and H
C_{OD}/C_{DO}	potential adjacency matrix
P	indication matrix for all complete entries of D
$\lambda_1;\lambda_2;\eta_1;\eta_2;\xi$	regularization parameters

Ni et al. [14] improved prediction accuracy by using social media events as auxiliary information, while Leng et al. [11] and Sun et al. [20] both take the transfer passenger flows for each station into consideration. However, none of the above studies construct a network-wide prediction system because the widely-used time series approaches, such as ARIMA [24], GP[31], neural network [23, 25, 26], are either computationally expensive or require lengthy off-line training periods that are difficult to adapt to large-scale network-wide flow forecasting. A few studies have explored movement patterns rather than flow predictions on metro networks, such as [13, 19]. But, again, no method addresses network-wide CFD prediction problems because these tasks present far greater complexity.

2.2 Online Non-negative Matrix Factorization

Ever since non-negative matrix factorization (NMF) was first proposed, it has gained extensive popularity as a solution for network-wide problems, such as in recommendation systems [8] and community detection systems [28]. Unlike traditional NMF, online NMF methods can consider temporal information to track potential changes as time passes [3, 6]. To date, ONMF has been successfully applied to several areas. Deng et al. [7] proposed a state-of-the-art latent space model that relies on the temporal and topological attributes of roads to predict network traffic speeds. Blondel et al. [2] presented an online passive-aggressive algorithm using NMF to solve a constrained optimization problem, and Wang et al. [22] used ONMF to efficiently handle very large-scale and/or streaming datasets. However, the existing ONMF methods are not yet able to directly solve CFD prediction problems. In this paper, we take advantage of ONMF to design two CFD prediction models that are tailored to adapt to different situations and can be applied directly.

3 PROBLEM STATEMENT

This section presents a formal definition of the CFD prediction problem in the Sydney Trains network. For ease of presentation, the main symbols used in this paper are summarized in Table 1.

In this CFD prediction problem, every origin-destination pair for every crowd flow needs to be recorded. The CFD network is defined as a directed graph $G = (V, E)$, where V is the set of vertexes and E is the set of edges. A vertex $v_i \subseteq V$ denotes the entrance or exit of a train station, and an edge $e(v_i, v_j)$ records an origin-destination pair from station v_i to v_j. Since our model focuses on both entrance and exit flows, the value of each edge $e(v_i, v_j)$ is associated with two observed flows $x(v_i, v_j)$ and $y(v_i, v_j)$, where $x(v_i, v_j)$ records the number of passengers that tapped on tickets at the i^{th} station and are going to the j^{th} station (i.e., an origin to destination pair, (OD)); $y(v_i, v_j)$ records the number of passengers that tapped off tickets at the i^{th} station and came from j^{th} station (i.e., a destination to origin pair, (DO)). Let $D_{OD} = (x_{ij})$ and $D_{DO} = (y_{ij})$ be the two corresponding matrices of G, x_{ij} and y_{ij} are represented by $x(v_i, v_j)$ and $y(v_i, v_j)$, respectively.

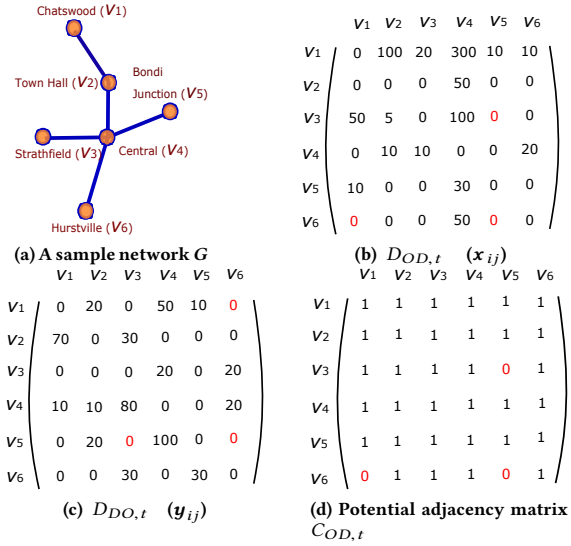


Figure 2: The topology example of Sydney trains network

The prediction interval is set to 15 minutes, as recommended in the relevant literature [5, 18, 20, 21], such that one day is divided into 96 snapshots. For each time interval t, we obtain $D_{OD,t}$ and $D_{DO,t}$ from the network G_t. For example, Figure 2 (a) shows a section of the Sydney Trains network G_t with six stations at one timestamp t. Figures 2 (b) and (c) show the two corresponding matrices $D_{OD,t}$ and $D_{DO,t}$. In Figure 2 (b), $x_{12} = 100$ means that 100 passengers tapped on at Chatswood (v_1) and they are going to Town Hall (v_2) . So, the sum of each row in $D_{OD,t}$ is the total number of people exiting each station at timestamp t. In Figure 2 (c), $y_{43} = 80$ means there are 80 passengers tapped off at Central (v_4) that came from Strathfield (v_3), so the sum of each row in $D_{DO,t}$ is the total number of exit crowd of each station at timestamp t.

In the real world, $D_{OD,t}$ and $D_{DO,t}$ are very sparse. Hence, even with a huge amount of historical data, a crowd flow may rarely occur between some stations. For simplicity, let us take Figure 2 (b) as an example. The positions red marked x_{35}, x_{65}, and x_{61} in $D_{OD,t}$ represent the routes from Hurstville to Chatswood and illustrate that passengers rarely choose these routes probably because of a much better alternative. Based on this real-life phenomenon, we can build the potential adjacency matrices for $D_{OD,t}$ and $D_{DO,t}$, denoted as $C_{OD,t}$ (as shown in Figure 2 (d)) and $C_{DO,t}$ at timestamp t, to reduce the *high computational complexity* of the problem. The position marked as 1 in $C_{OD,t}$ means the i^{th} and j^{th} stations are connected, while a 0 indicates a potential disconnected.

Unlike $D_{DO,t}$, in real-time system, it is impossible to construct an exact $D_{OD,t}$ for present timestamp t. We refer to this as the

incomplete data challenge. $D_{DO,t}$ collects exit flow data; thus it is easy to determine where a passenger came from. However, $D_{OD,t}$ collects entrance flow data and the number of entries will grow over the next several timestamps until all passengers have reached their destination and tapped off, as illustrated in Figure 3.

Figure 3: Data collection for $D_{OD,t}$

Therefore, several new strategies are required as follows. **(1)** A measurement method needs to be developed to ensure that a complete set of data has been collected. **(2)** Given the real-time dynamic sequences of CFD, D_{OD} ($D_{OD,1}$, $D_{OD,2}$, $\cdots$, $D_{OD,T}$) and D_{DO}($D_{DO,1}$, $D_{DO,2}$, $\cdots$, $D_{DO,T}$), we forecast the short-term changes in CFD, i.e., forecasting $D_{OD,T+1}$ and $D_{DO,T+1}$ on-the-fly.

We emphasize that $D_{OD,T}$ and $D_{DO,T}$ matrices are two separate problems (they are related in daily patterns but not in 15 min patterns), the former records the number of passengers that tapped on tickets at the i^{th} station and are going to the j^{th} station, termed as entrance CFD prediction; the latter records the number of passengers that tapped off tickets at the i^{th} station and came from j^{th} station, termed as exit CFD prediction. Due to the page limitation, in this paper we have only presented the optimization strategy for entrance CFD prediction ($D_{OD,T}$) because both can be addressed in a similar way.

4 THE ONMF-AO MODEL

Our prediction model is built on NMF, so Section 4.1 discusses how a basic NMF model can be applied in Sydney Trains Network. In Section 4.2, we propose the ONMF-AO model, which incorporates the average optimization of transition patterns. Lastly, Section 4.3 outlines how historical guidance is used to improve and complete the model.

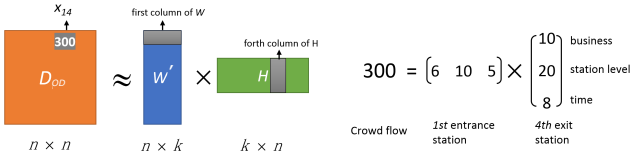
4.1 NMF in The CFD Model

In CFD prediction problems, the non-negative matrix factorization method decomposes the crowd flow matrix $D_{OD} \in \mathbb{R}_+^{n\times n}$ into two matrices $W \in \mathbb{R}_+^{k\times n}$ and $H \in \mathbb{R}_+^{k\times n}$, where W and H represent the latent spaces. Each column in these matrices represents k attributes of corresponding entrance and exit stations. As we mentioned in Section 3, here we only use D_{OD} as an example because $D_{DO} \in \mathbb{R}_+^{n\times n}$ has a similar description to D_{OD}. The interaction between these attributes determines the crowd flow between the entrance and exit stations. Therefore, the basic CFD model can be described as the following optimization objective:

$$\arg\min_{W\geq 0, H\geq 0} L_0 = ||D_{OD} - W^{'}H||_F^2 \tag{1}$$

where $W^{'}$ is the transposed matrix of W.

Figure 4 (a) illustrates how an NMF model can be used to solve a CFD problem in one snapshot. For example, CFD (x_{14}) is associated



(a) NMF model of CFD **(b) interaction of latent attributes to estimate crowd flow**

Figure 4: The latent space example of Sydney trains network

with two sets of latent attributes, which can be hypothesized as a business area, the station level, and a time (rush degree) when setting k=3, as shown in Figure 4 (b). Note that the dimension of latent space k is a hyper-parameter, Figure 4 (b) gives a reasonable assumption for an example to reveal the possible attributes in the latent spaces .

4.2 Prediction Using an Average Optimization Strategy

In real life, CFDs are continually changing over time, giving rise to the dynamics in $D_{OD,t}/D_{DO,t}$ as well as their decomposed matrices in W_t and H_t. Hence, to achieve accurate predictions, these transition patterns from previous snapshots to the next need to be captured.

The transition matrices $A \in \mathbb{R}_+^{k\times k}$ and $B \in \mathbb{R}_+^{n\times n}$ reveal the global potential tendencies of $W^{'}$ and H over a certain time window T (i.e., T consecutive timestamps) based on an average optimization strategy. We chose this optimization method to avoid interference by short-term noise, such as timetable changes, traffic conditions around the stations, and so forth. Therefore, the global tendency pattern can be thought of as a representation of the stable CFD changes from timestamp 1 to timestamp T, i.e., $\sum_{t=1}^{T} W_t^{'} = W_{t-1}^{'}A$ and $\sum_{t=1}^{T} H_t = H_{t-1}B$, where A and B approximate the average changes in the time window.

Our model includes two components, the entrance CFD ($D_{OD,t}$) and the exit CFD ($D_{DO,t}$) prediction. For simplicity, we have only presented the optimization strategy for $D_{OD,t}$ because both are addressed in a similar way. Thus, the prediction model based on the average optimization strategy is shown as Equation 2.

$$\begin{aligned}\arg\min_{W_t,H_t,A,B} L_1 = &\sum_{t=1}^{T} ||P_{OD,t} \odot (D_{OD,t} - W_t^{'}H_t)||_F^2 + \\ &\lambda_1 \sum_{t=2}^{T} (||W_t^{'} - W_{t-1}^{'}A||_F^2 + ||H_t - H_{t-1}B||_F^2)\end{aligned} \tag{2}$$

where λ_1 is the regularization parameter; $\odot$ is the Hadamard product (entrywise product) operator; and $P_{OD,t}$ is the indication matrix for all the complete entries of $D_{OD,t}$. We discuss how to derive $P_{OD,t}$ in the following.

In the entrance flow prediction problem, we only set $P_{OD,t(ij)}$ = 1 ($1 \leq i,j \leq n$) if the time horizon between t and present time T is sufficient for the vast majority of people to have arrived at their destination. Now, let us discuss how to derive $P_{OD,t}$ in the entrance problem.

Given the ***incomplete data challenge*** discussed in Section 3, passengers with the same destination will tap off their Opal card

tickets at different times even though they tapped on at the same place in the same timestamp. In fact, as time passes, the number of entries in $D_{OD,t}$ will only complete as the last passenger completes their journey and taps off at their destination. So, how can we be sure that the data collected are complete? Based on [9], the travel times for each origin-destination pair in each timestamp can be assumed to be close to the normal distribution, i.e., $Z_{q,t} \sim N(\mu_{q,t}, \sigma^2_{q,t})$. $Z_{q,t}$ denotes the travel times for one origin-destination pair at timestamp t (q^{th}, $q \in$ all potential CFD) (e.g., all historical travel times the origin to a given destination commencing from time t). Then, if the time span is greater than $\mu_{q,t} + 2\sigma_{q,t}$, we have approximately 98% confidence that all passengers have arrived at their destination based on the normal distribution.

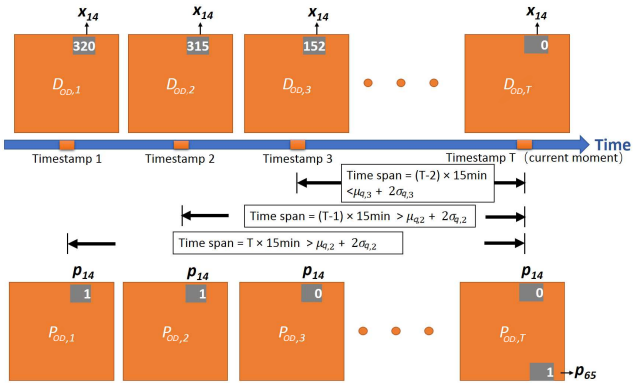


Figure 5: An example of building indication matrix $P_{OD,t}$

For example, Figure 5 illustrates how to build the indication matrix $P_{OD,t}$. $D_{OD,1}$ is the entrance matrix that we had collected 320 passengers that tapped on their tickets at station v_1 at timestamp 1 (the first timestamp of the window) and then tapped off at station v_4 until T. There are no exit records for the current moment matrix $D_{OD,T}$. 152 passengers exited at timestamp 3 but the time span between 3 and T is smaller than $\mu_{q,3} + 2\sigma_{q,3}$, which does not provide enough confidence to indicate that 152 reflects complete data. So, we set p_{14}=0 in $P_{OD,3}$. However, the collected data at timestamps 1 and 2 (320 and 315) have enough time spans indicating these data is complete with a confidence level above 98% (then set $p_{14} = 1$ in $P_{OD,1}$ and $P_{OD,2}$). Further, if the origin-destination pair is potentially disconnected based on the potential adjacency matrix $C_{OD,t}$, the corresponding position in $P_{OD,t}$ will be set to 1 on the assumption that the latent attributes of disconnected stations for each origin-destination pair are orthogonal. For instance, we set p_{65}=1 in $P_{OD,T}$ because we assume the latent attributes of entrance station v_6 and exit station v_5 are orthogonal.

4.3 Incorporation with Historic Guidance

In the Sydney Trains network, CFDs are usually very stable and have a strong daily periodic property, especially on weekdays. that means our prediction objective $D_{OD,T+1}$ is similar to the history matrix $D^h_{OD,T+1}$ thanks to daily periodicity. Based on this phenomenon, we can incorporate historic guidance into our model. Assume that W^h_{T+1} and H^h_{T+1} are two latent matrices of the next snapshot $D^h_{OD,T+1}$ of history. We seek to learn the transition matrices A and B from now to the next timestamp by consulting these two historic matrices W^h_{T+1} and H^h_{T+1}. More importantly, historic guidance brings a strong benefit to sharpening transformation from rush hour to non-rush hour.

The historic matrices can be easily learned by Equation 2 with the slight modification, of replacing $D_{OD,t}$, W_t and H_t with $D^h_{OD,t}$, W^h_t and H^h_t, respectively, and setting T to $T+1$. Let all entries in the indication matrix $P_{OD,t}$ is 1, because all historical data are complete.

To optimize the model and avoid over-fitting problems, we add a constrain for W_t to limit the search space. Considering all the above strategies, the final loss function of our ONMF-AO prediction model is defined as:

$$\begin{aligned} \underset{W_t,H_t,A,B}{\arg\min}\ L_{AO} &= L_1 + \lambda_2(||W^{h'}_{T+1} - W'_T A||^2_F + ||H^h_{T+1} - H_T B||^2_F) \\ s.t. \qquad & ||W_t||^2_F = \xi \quad ,\forall t = 1, \cdots, T. \end{aligned} \tag{3}$$

where λ_2 is the regularization parameter; W_T and H_T are the latent matrices when $t = T$; and ξ is an empirical parameter to constrain overall change of W_t.

The latent matrices W_t, H_t, A and B can be learned through Equation 3, and the equation for predicting the next snapshot $D_{OD,T+1}$ is $D_{OD,T+1} = (W'_T A)(H_T B)$.

4.4 Learning and Prediction with ONMF-AO

As Equation 3 is a non-convex problem, we use the multiplicative update strategy [10] and the proximal gradient method [12, 15] to find a local optimal solution. We initialize latent space matrices (W_t and H_t) in each timestamp with corresponding historical values, which were derived from historic data, to ensure a better solution. Such initialization is reasonable under the assumption that model parameters won't change much compared to its historical values. The update rules for W_t, H_t, A, and B are presented in Equations 4 - 9.

$$W_t = \mathbf{prox}(W_t \odot \frac{H_t(P_{OD,t} \odot D_{OD,t})' + \lambda_1(A' W_{t-1} + A W_{t+1}) + \hat{\lambda_2}(A W^h_{T+1})}{H_t(P_{OD,t} \odot (W'_t H_t))' + \lambda_1(W_t + A A' W_t) + \hat{\lambda_2} A A' W_T}) \tag{4}$$

$$H_t = H_t \odot \frac{W_t(P_{OD,t} \odot D_{OD,t}) + \lambda_1(H_{t-1} B + H_{t+1} B') + \hat{\lambda_2}(H^h_{T+1} B')}{W_t(P_{OD,t} \odot (W'_t H_t)) + \lambda_1(H_t + H_t B B') + \hat{\lambda_2} H_T B B'} \tag{5}$$

where $\mathbf{prox}(\cdot)$ is the proximal operator defined as:

$$\mathbf{prox}(W_t) = \xi W_t / ||W_t||^2_F, \tag{6}$$

and $\hat{\lambda_2}$ is given by:

$$\hat{\lambda_2} = \begin{cases} \lambda_2, & t = T \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

The update equations for A and B are:

$$A = A \odot \frac{\lambda_1 \sum_{t=1}^{T} W_{t-1} W_t^{'} + \lambda_2 W_T W_{T+1}^{h'}}{\lambda_1 \sum_{t=1}^{T} W_{t-1}(W_{t-1}^{'} A) + \lambda_2 W_T (W_T^{'} A)} \tag{8}$$

$$B = B \odot \frac{\lambda_1 \sum_{t=1}^{T} H_{t-1}^{'} H_t + \lambda_2 H_T^{'} H_{T+1}^{h}}{\lambda_1 \sum_{t=1}^{T} H_{t-1}^{'}(H_{t-1} B) + \lambda_2 H_T^{'}(H_T B)} \tag{9}$$

Based on the above update equations, the iterative learning and prediction process for ONMF-AO is summarized in Algorithm 1.

Algorithm 1: ONMF-AO Prediction

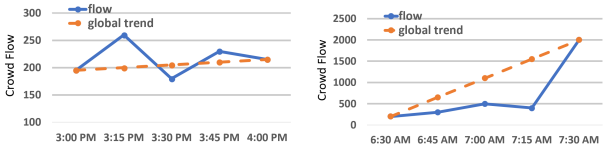
INPUT: present CFD ($D_{OD,1}, \cdots, D_{OD,T}$); historic guidance W_{T+1}^{h}, H_{T+1}^{h}; error threshold ε; maximum iteration $Iter$
OUTPUT: next snapshot $D_{OD,T+1}$
(1) initialize W_t, H_t, A and B
(2) **While** $i < Iter$
(3) **If** $L_{AO,i}$-$L_{AO,i+1} > \varepsilon$
(4) **For** t= 1:T do
(5) update W_t and H_t **By** Equation 4 - 5
(6) update A and B **By** Equation 8 and 9
(7) **Else Break**
(8) **Return** $D_{OD,T+1}$ **By** $D_{OD,T+1} = (W_T^{'} A)(H_T B)$

5 THE MODEL FOR SUDDEN FLOW CHANGES

In this section, we propose a second ONMF-based model,called ONMF-MR, to handle scenarios where sudden changes flow occur.

5.1 Improved Model ONMF-MR

ONMF-AO relies on an average optimization strategy, which performs better when forecasting stable passenger flows, as shown in Figure 6 (a). This is because the global potential tendencies can filter non-recurrent noises, especially in the off-peak times. However, a sudden increase flow would appear during the rush hour, e.g., between 7:15 to 7:30 AM in Figure 6 (b), the crowd flow shoots upward. However, the ONMF-AO is insensitive to this situation because it consider the average trends of a given number of previous timestamps.



(a) in non-rush hour **(b) in rush hour**

Figure 6: An example CFD from Hurstville to Central

Therefore, to improve the prediction accuracy of the model when faced with sudden or drastic changes in flow, we designed a second ONMF model, called ONMF-MR. The improved model aims to learn the most recent CFD trends. The transition matrices A and B are disassembled into each timestamp, i.e., learning A_t and B_t from one snapshot to the next. Motivated by this idea, we modify the optimization function L_1 in Equation 2 to tackle the most recent dynamic trends as follows:

$$\underset{W_t, H_t, A_t, B_t}{\arg\min} \; L_2 = \sum_{t=1}^{T} ||P_{OD,t} \odot (D_{OD,t} - W_t^{'} H_t)||_F^2 + \frac{\eta_1}{2} \sum_{t=2}^{T} (||W_t^{'} - W_{t-1}^{'} A_{t-1}||_F^2 + ||H_t - H_{t-1} B_{t-1}||_F^2) \tag{10}$$

where η_1 is the regularization parameter.

After incorporating the historic guidance described in Section 4.3, the final loss function of ONMF-MR is

$$\underset{W_t, H_t, A, B}{\arg\min} \; L_{MR} = L_2 + \eta_2 (||W_{T+1}^{h'} - W_T^{'} A_T||_F^2 + ||H_{T+1}^{h} - H_T B_T||_F^2)$$
$$s.t. \qquad ||W_t||_F^2 = \xi \quad , \forall t = 1, \cdots, T. \tag{11}$$

where η_2 is the regularization parameter; A_T and B_T are transition matrices when $t = T$.

5.2 Learning and Prediction by ONMF-MR

The update rules of W_t, H_t, A_t, and B_t are presented in Equations 12 to Equation 15.

$$W_t = \textbf{prox}(W_t \odot \frac{H_t (P_{OD,t} \odot D_{OD,t})^{'} + \eta_1 (A_{t-1}^{'} W_{t-1} + A_t W_{t+1}) + \hat{\eta_2} (A_T W_{T+1}^{h})}{H_t (P_{OD,t} \odot (W_t^{'} H_t))^{'} + \eta_1 (W_t + A_t A_t^{'} W_t) + \hat{\eta_2} A_T A_T^{'} W_T}) \tag{12}$$

where **prox**$(\cdot)$ is the same proximal operator as in Equation 6.

$$H_t = H_t \odot \frac{W_t (P_{OD,t} \odot D_{OD,t}) + \eta_1 (H_{t-1} B_{t-1} + H_{t+1} B_t^{'}) + \hat{\eta_2} (H_{T+1}^{h} B_T^{'})}{W_t (P_{OD,t} \odot (W_t^{'} H_t)) + \eta_1 (H_t + H_t B_t B_t^{'}) + \hat{\eta_2} H_T B_T B_T^{'}} \tag{13}$$

$$A_t = A_t \odot \frac{\eta_1 W_t W_{t+1}^{'} + \hat{\eta_2} W_T W_{T+1}^{h'}}{\eta_1 W_t (W_t^{'} A_t) + \hat{\eta_2} W_T (W_T^{'} A_T)} \tag{14}$$

$$B_t = B_t \odot \frac{\eta_1 H_t^{'} H_{t+1} + \hat{\eta_2} H_T^{'} H_{T+1}^{h}}{\eta_1 H_t^{'} (H_t B_t) + \hat{\eta_2} H_T^{'} (H_T B_T)} \tag{15}$$

Equation 12 - 15 satisfy:

$$\hat{\eta_2} = \begin{cases} \eta_2, & t = T \\ 0, & \text{otherwise} \end{cases}$$

Based on the above update equations, the iterative learning and prediction process for ONMF-MR is summarized in Algorithm 2.

Time complexity and convergence. we briefly discuss the time complexity and convergence of two proposed models ONMF-AO and ONMF-MR. Equations 4-9 and Equations 12-15 clearly present that the time complexity is governed by matrix multiplication operations in each iteration. Therefore, the time complexity of both two models is $O(Tn^2k)$. In terms of convergence, Algorithm 1 and 2 are guaranteed to converge as the proof shown in previous ONMF-based works [2, 7, 10].

Algorithm 2: ONMF-MR Prediction

INPUT: present CFD ($D_{OD,1}$, $\cdots$, $D_{OD,T}$); historic guidance W^h_{T+1}, H^h_{T+1}; error threshold ε; maximum iteration $Iter$
OUTPUT: next snapshot $D_{OD,T+1}$
(1) initialize W_t, H_t, A_t and B_t
(2) **While** i < $Iter$
(3) **If** $L_{AO,i}$-$L_{AO,i+1}$> ε
(4) **For** t= 1:T do
(5) update W_t, H_t, A_t and B_t **By** Equation 12 - 15
(6) **Else Break**
(7) **Return** $D_{OD,T+1}$ **By** $D_{OD,T+1} = (W'_T A_T)(H_T B_T)$

5.3 A Hybrid Strategy

Given that ONMF-AO and ONMF-MR have been specifically designed to perform well in two distinct scenarios (one with stable flows and the other with drastic changes to flows). We can build a hybrid model, ONMF-H, which integrates both ONMF-AO and ONMF-MR, as shown in Algorithm 3.

Considering current moment T, the ground-truth of DO CFD for the last timestamp *T-1* (matrix $D_{DO,T-1}$) is available. Therefore, the ONMF-AO model predicts the current timestamp if the accuracy of ONMF-AO at the last timestamp is greater than ONMF-MR, and vice versa.

However, due to the *incomplete data challenge*, we cannot obtain the complete ground-truth for predicted OD CFD matrix ($D_{OD,T-1}$) at timestamp T, but only have the total number of entrance passengers for each station at timestamp T-1. Hence, we compare the sum of each row in $D_{OD,T-1}$ with ground-truth, and use this accuracy as the selection criteria for the next prediction.

Algorithm 3: Hybrid Model — ONMF-H

(1) Choose different Accuracy criteria for different prediction problems as discussed in Section 5.3.
(2) **While** need prediction
(3) **If** Last_Accuracy (Algorithm 1) > Last_Accuracy (Algorithm 2)
(4) Using the result of Algorithm 1 for next timespan prediction; Running Algorithm 1 and 2 in parallel
(5) **Else**
(6) Using the result of Algorithm 2 for next timespan prediction; Running Algorithm 1 and 2 in parallel
(7) **Until** end prediction

6 EXPERIMENTS

In this section, we report on the experiments conducted on the Opal card transaction data of the Sydney Trains network. All algorithms were implemented in Matlab, and run on a virtual Linux machine with 20 CPUs and 64GB memory.

6.1 The Dataset

The dataset is a large-scale, real-world dataset provided by Transport NSW that contains the transactional data pertaining to the Sydney Trains network. After data cleaning, the dataset included over 30 million records covering 178 stations between 7 Nov 2016 and 11 Dec 2016. We used the records between 28 Nov. 2016 and 11 Dec. 2016 as our test data; the remaining records were used as the training data.

It's remarkable that the proposed methods are online models. For example, when crowd flow at timestamp T+1 is to be predicted, all data at timestamp T can be regarded as training data, and data at timestamp T+1 is used for validation. With the time going on, data at T+1 timestamp becomes the training data and T+2 becomes validation data. Finally we go through all the training data, and get the average error on validation data. With different hyper-parameters, we got different errors, and we choose hyper-parameters with lower error to be used in test model.

6.2 Baselines & Measures & Parameters

Baselines. We compare all proposed models, ONMF-AO, ONMF-MR, and ONMF-H with the following five baselines. All parameters of proposed and baselines are optimized by the grid search method.

• **HA:** CFD predictions are made using the historical average method in each time span. That is, all historical time intervals from 8:00 am to 8:15 am on Mondays are used to make a prediction about the same timestamp on the test Monday.

• **ONMF:** We removed the historic guidance in our models to test the performance of a traditional ONMF-based method.

• **LSM-RN-All:** A state-of-the-art ONMF-based model for solving traffic network prediction problem[7].

• **SARIMA:** the well-known seasonal auto-regressive integrated moving average model — a linear regression model for forecasting future values in a time series.

• **GPR:** Gaussian process regression (GPR) is a stochastic process model that can be used to capture spatiotemporal patterns but at a heavy computational cost [17].

Measures. We measure our methods by Mean Absolute Error (MAE) and Mean Relative Error (MRE) which are commonly used to evaluate time series accuracy [27].

$$MAE = \frac{\sum_{i=1}^{m} |v_i - \hat{v}_i|}{m}, \qquad MRE = \frac{\sum_{i=1}^{m} |v_i - \hat{v}_i|}{\sum_{i=1}^{m} v_i},$$

where $\hat{v}_i$ is a prediction and v_i is the ground truth; m is the number of prediction flows.

Initialization. We initialize latent space matrices of proposed models and ONMF-based baselines in each timestamp with corresponding historical values, which were derived from historic data, to ensure a better solution. Such initialization is reasonable under the assumption that model parameters won't change much compared to its historical values.

Parameter Settings. The parameters used to achieve the best performance in different application scenarios are shown in Table 2. In addition, we set the proximal gradient parameter ξ to 30 after a line search process.

6.3 Results on the Sydney Train Network

The first set of experiments was designed to assess performance across the entire Sydney Trains network. The results appear in Figure 7, including the comparisons for entrance/exit CFDs (i.e., OD and DO CDFs) between rush and non-rush periods and between

Table 2: Symbol Description

ONMF-AO	k	T	λ_1	λ_2
Entrance CFD in Weekdays	70	6	2^6	2^{13}
Exit CFD in Weekdays	70	6	2^5	2^{15}
Entrance CFD in Weekends	70	6	2^6	$2^{6.5}$
Exit CFD in Weekends	70	6	2^2	2^2
ONMF-MR	k	T	η_1	η_2
Entrance CFD in Weekdays	70	3	$2^{1.5}$	2^5
Exit CFD in Weekdays	70	3	$2^{5.5}$	2^{18}
Entrance CFD in Weekends	70	3	$2^{0.5}$	2^2
Exit CFD in Weekends	70	3	$2^{3.5}$	2^4

weekdays and weekends. We have only included the top four methods to keep the images clear, as the other baselines performed far worse than these approaches. The lines of Hybrid model overlap with the best one of ONMF-AO and ONMF-ND because the Hybrid model will take advantages of the first two models in different scenarios which cause a bad presentation in Figures. Due to the above reason, we remove the lines of Hybrid model in Figures and place them to the overall results in Table 3.

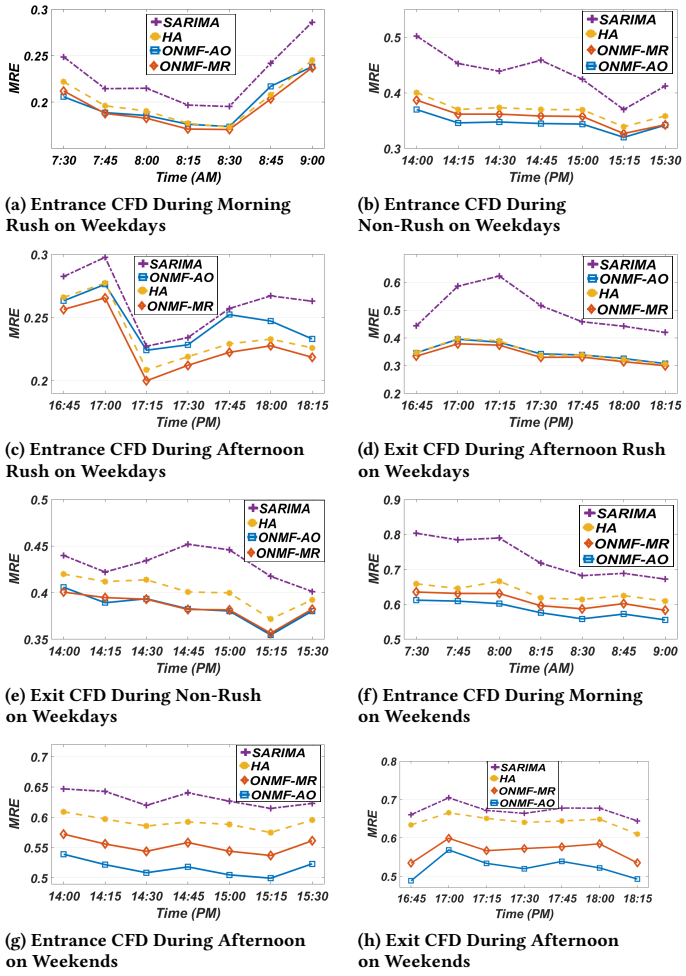
As the weekday tests drawn in Figure 7 (a)-(e), it is clearly shown that ONMF-AO performed better during non-rush times, while ONMF-MR performed better during rush times in both the entrance and the exit CFD prediction tasks. ONMF-AO achieved the best results on weekends because passengers flows are more stable throughout the day (as shown in Figure 7 (f)-(h)). The simple strategy HA performed well because CFDs are highly stable across the Sydney Trains network, especially on weekdays. However, predictions based on historical data may be deceiving when faced with non-recurring events, as discussed in Section 1. The regression methods, SARIMA and GPR, are sensitive to missing data [7], but SARIMA considers seasonal properties, which improved accuracy to some degree. The traditional ONMF and LSM-RN-All did not perform well because they do not consider historic trends. More comparisons are shown in Section 6.5.

6.4 Results for the Major Stations

We further evaluated the effectiveness of each method on the major stations in the Sydney Trains network. The top 20 stations based on throughput capacity were selected as the major stations. This experiment removes any interference created by smaller stations to assess the models' performance more deeply, as shown in Figure 8. The results lead to similar conclusions to the first experiment. Our models provided better results than the other baselines. ONMF-MR achieved a slightly better performance than HA in peak times on weekdays, as shown in Figure 8 (a) (c) and (d). ONMF-AO was the best prediction method for non-peak times on weekdays and all times on the weekend, as shown in Figure 8 (b) and (e) - (h).

6.5 Overall Results

Table 3 presents the average errors of all test methods for each timestamp between 6.00am and 10.00pm. ONMF-AO outperformed the other baselines on weekends because sudden changes in flow are



(a) Entrance CFD During Morning Rush on Weekdays

(b) Entrance CFD During Non-Rush on Weekdays

(c) Entrance CFD During Afternoon Rush on Weekdays

(d) Exit CFD During Afternoon Rush on Weekdays

(e) Exit CFD During Non-Rush on Weekdays

(f) Entrance CFD During Morning on Weekends

(g) Entrance CFD During Afternoon on Weekends

(h) Exit CFD During Afternoon on Weekends

Figure 7: CFD Prediction on the Sydney Trains Network

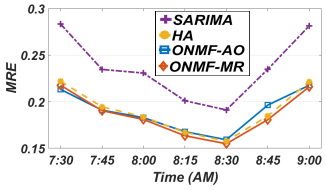
rare. While on weekdays, ONMF-AO and ONMF-MR each provide better performance depending on the circumstance; hence, the hybrid model ONMF-H achieved the best results on weekdays.

Even though the regression methods, such as SARIMA and GPR, can be used to solve time series problems, we do not recommend them for solving network-wide CFD prediction problems due to the huge re-training cost and the high complexity associated with considering each potential passenger flow. Nor do we recommend the latest approaches in addressing traffic prediction problems on networks, such as LSM-RN-All [7] and ONMF [2], as these methods did not produce good results. We attribute this to the strong periodicity of CFD prediction problems. LSM-RN-All and ONMF were able to capture variations in CFD trends within small time windows but do not incorporate historic guidance, so performance suffers.

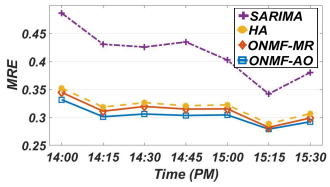
Overall, the two purpose-built ONMF models, ONMF-AO and ONMF-MR, have their own advantages for network-wide CFD predictions for Sydney Trains. The hybrid model provides the best results for weekdays and ONMF-AO is the most outstanding method for forecasting on weekends.

Table 3: Comparison Among Different Methods

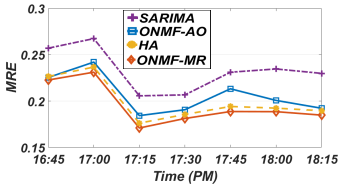
Methods	Entrance CFD on Weekdays		Exit CFD on Weekdays		Entrance CFD on Weekends		Exit CFD on Weekends		Average	
	MAE	MRE	MAE	MRE	MAE	MRE	MAE	MRE	MAE	MRE
ONMF-AO	1.5932	0.3142	1.9540	0.3748	**1.9491**	**0.5531**	**2.1559**	**0.5901**	1.9131	0.4581
ONMF-MR	1.5981	0.3182	1.9162	0.3694	2.0426	0.5784	2.2199	0.6048	1.9442	0.4677
ONMF-H	**1.5820**	**0.3139**	**1.9144**	**0.3690**	1.9847	0.5536	2.1595	0.5902	**1.9102**	**0.4567**
HA	1.6498	0.3285	1.9727	0.3819	2.1793	0.6148	2.4340	0.6598	2.0590	0.4963
LSM-RN-All	6.2915	1.1543	3.5580	0.6722	2.1077	0.6031	2.9429	0.8024	3.7250	0.8080
ONMF	4.0940	0.7735	2.5978	0.4879	2.1460	0.6117	3.3827	0.9165	3.0550	0.6974
SARIMA	2.0005	0.4051	2.2514	0.4544	2.3932	0.6851	2.6596	0.7349	2.3262	0.5699
GPR	5.3813	1.1507	5.3324	1.1135	2.4212	0.6854	2.6866	0.7285	3.9554	0.9195



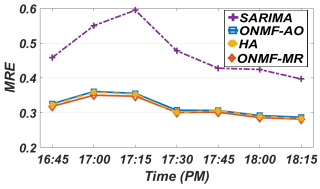
(a) Entrance CFD During Morning Rush on Weekdays



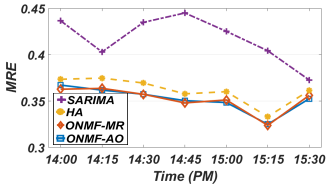
(b) Entrance CFD During Non-Rush on Weekdays



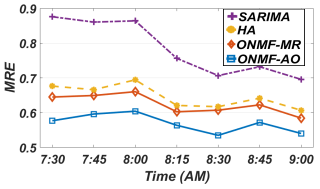
(c) Entrance CFD During Afternoon Rush on Weekdays



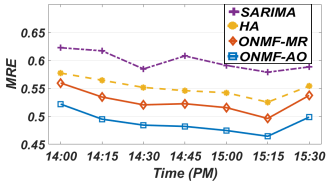
(d) Exit CFD During Afternoon Rush on Weekdays



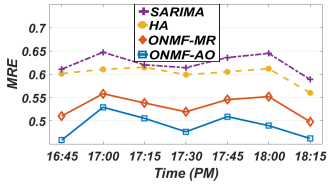
(e) Exit CFD During Non-Rush hour on Weekdays



(f) Entrance CFD During Morning on Weekends

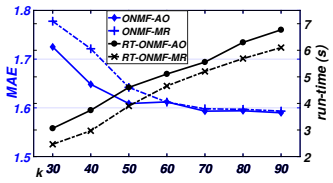


(g) Entrance CFD During Afternoon on Weekends

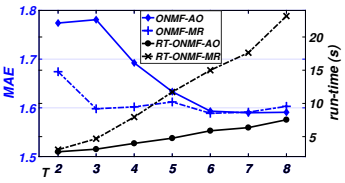


(h) Exit CFD During Afternoon on Weekends

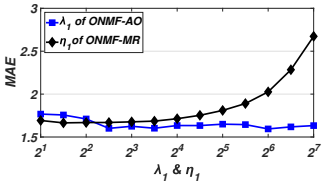
Figure 8: CFD Prediction on the Major Stations



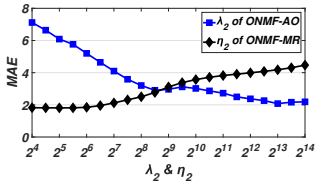
(a) Factor k



(b) Factor T



(c) Factor λ_1&η_1



(d) Factor λ_2&η_2

Figure 9: Effect of Parameters

6.6 The Sensitivity of Parameters

In this section, we evaluate the performance of our methods by varying the parameters of our model. Due to space limitations, we have only shown the experimental results for the entrance CFD on weekdays.

Figure 9 (a) shows the different performances and running times (RT) with a varying setting for k. Both ONMF-AO and ONMF-MR achieved better results but longer running times as the number of latent attributes increased. We set k to 70 for a good balance between time-consuming and accuracy.

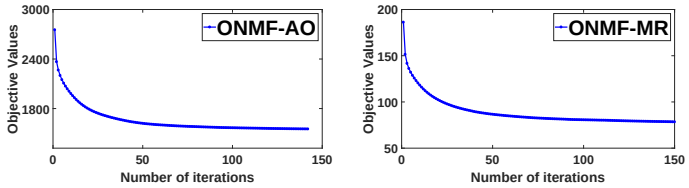
Figure 9 (b) plots the different performances and running times when changing T. ONMF-AO had a minimum MAE when T=6. ONMF-MR had a good result when $T \geq 3$. thus, we suggest setting T to 3 the running time appears to grow rapidly as T increases.

Figure 9 (c) reveals the effect of varying λ_1 and η_1. These two parameters determine the strength of the current flow trends to use. λ_1=2^6 and η_1=$2^{1.5}$ yielded the best results for ONMF-AO and ONMF-MR, respectively.

Figure 9 (d) shows the impacts of varying λ_2 and η_2 which effect the power of historic guidance. ONMF-AO required more historic guidance than ONMF-MR to achieve good results. The best values of λ_2 and η_2 were 2^{13} and 2^5, respectively.

6.7 Scalability

We used a scalability test to evaluate the efficiency of each of the different methods. We only recorded the longest time-consumption for ONMF-AO and ONMF-MR because these models comprise the hybrid method. Table 4 shows the results of the training and forecasting times for one prediction step across the entire network.



(a) ONMF-AO (b) ONMF-MR

Figure 10: Converge rate

The regression methods, SARIMA and GPR, required an enormous amount of re-training time, which is difficult to implement in a real-time system. However, both methods did generate predictions very quickly. ONMF-AO, ONMF-MR, LSM, and the traditional version of ONMF had similar running times and did not require additional re-training time. LSM and ONMF were faster at prediction than ONMF-AO, ONMF-MR but show a lower accuracy. The scalability test demonstrates that our models, ONMF-AO and ONMF-MR, perform well on large-scale network problems, taking approximately five seconds for each prediction step.

Convergence. Figures 10 (a) and (b) show the convergence trends of iterative models ONMF-AO and ONMF-MR. Both Algorithms converge into a local solution in a small amount of iterations.

Table 4: Scalability Test. ONMF-AO/ONMF-MR completed each prediction step in a reasonable time span (about 5 seconds) with the highest accuracy.

	ONMF-AO/MR	LSM	ONMF	SARIMA	GPR
re-train(s)	-	-	-	156.55	399.82
pred.(s)	5.01	3.69	4.96	0.68	0.36
train+pred.(s)	5.01	3.69	4.96	157.23	400.18

7 CONCLUSIONS

In this paper, we propose two data-driven models for CFD prediction on the Sydney Trains rail network. The first model, called ONMF-AO, is based on average optimization and historic guidance and captures the dynamic changes in latent attributes over time. To improve prediction accuracy, we further designed another ONMF model, called ONMF-MR, to tackle sudden changes in CFDs. Intensive experiments show that our proposed methods outperform several baselines. A hybrid strategy, which combines both ONMF-AO and ONMF-MR, achieves the best results on weekdays, and ONMF-AO is the most outstanding method for weekend predictions.

REFERENCES

[1] Gowtham Atluri, Anuj Karpatne, and Vipin Kumar. 2017. Spatio-Temporal Data Mining: A Survey of Problems and Methods. *arXiv:1711.04710* (2017).

[2] Mathieu Blondel, Yotaro Kubo, and Ueda Naonori. 2014. Online passive-aggressive algorithms for non-negative matrix factorization and completion. In *Artificial Intelligence and Statistics*. 96–104.

[3] Bin Cao, Dou Shen, Jian-Tao Sun, Xuanhui Wang, Qiang Yang, and Zheng Chen. 2007. Detect and Track Latent Factors with Online Nonnegative Matrix Factorization.. In *IJCAI*, Vol. 7. 2689–2694.

[4] Irina Ceapa, Chris Smith, and Licia Capra. 2012. Avoiding the crowds: understanding tube station congestion patterns from trip data. In *Proceedings of the ACM SIGKDD international workshop on urban computing*. ACM, 134–141.

[5] Mu-Chen Chen and Yu Wei. 2011. Exploring time variants for short-term passenger flow. *Journal of Transport Geography* 19, 4 (2011), 488–498.

[6] Freddy Chong Tat Chua, Richard J Oentaryo, and Ee-Peng Lim. 2013. Modeling temporal adoptions using dynamic matrix factorization. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*. IEEE, 91–100.

[7] Dingxiong Deng, Cyrus Shahabi, Ugur Demiryurek, Linhong Zhu, Rose Yu, and Yan Liu. 2016. Latent Space Model for Road Networks to Predict Time-Varying Traffic. In *Proceedings of the 22nd ACM SIGKDD, August 13-17, 2016*. 1525–1534.

[8] Sheng Gao, Hao Luo, Da Chen, Shantao Li, Patrick Gallinari, and Jun Guo. 2013. Cross-domain recommendation via cluster-level latent factor model. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 161–176.

[9] Aude Hofleitner, Ryan Herring, and Alexandre Bayen. 2012. Probability distributions of travel times on arterial networks: a traffic flow and horizontal queuing theory approach. In *91st Transportation Research Board Annual Meeting*.

[10] Daniel D Lee and H Sebastian Seung. 2001. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*. 556–562.

[11] Biao Leng, Jiabei Zeng, Zhang Xiong, Weifeng Lv, and Yueliang Wan. 2013. Probability tree based passenger flow prediction and its application to the Beijing subway system. *Frontiers of Computer Science* 7, 2 (2013), 195–203.

[12] Kun-Yu Lin, Chang-Dong Wang, Yu-Qin Meng, and Zhi-Lin Zhao. 2017. Multi-view unit intact space learning. In *International Conference on Knowledge Science, Engineering and Management*. Springer, 211–223.

[13] Xiaolei Ma, Yao-Jan Wu, Yinhai Wang, Feng Chen, and Jianfeng Liu. 2013. Mining smart card data for transit riders' travel patterns. *Transportation Research Part C: Emerging Technologies* 36 (2013), 1–12.

[14] Ming Ni, Qing He, and Jing Gao. 2017. Forecasting the subway passenger flow under event occurrences with social media. *IEEE Transactions on Intelligent Transportation Systems* 18, 6 (2017), 1623–1632.

[15] Neal Parikh, Stephen Boyd, et al. 2014. Proximal algorithms. *Foundations and Trends® in Optimization* 1, 3 (2014), 127–239.

[16] Marie-Pier Pelletier, Martin Trépanier, and Catherine Morency. 2011. Smart card data use in public transit: A literature review. *Transportation Research Part C: Emerging Technologies* 19, 4 (2011), 557–568.

[17] Carl Edward Rasmussen and Christopher KI Williams. 2006. *Gaussian processes for machine learning*. Vol. 1. MIT press Cambridge.

[18] Lijun Sun, Jian Gang Jin, Der-Horng Lee, Kay W Axhausen, and Alexander Erath. 2014. Demand-driven timetable design for metro services. *Transportation Research Part C: Emerging Technologies* 46 (2014), 284–299.

[19] Lijun Sun, Yang Lu, Jian Gang Jin, Der-Horng Lee, and Kay W Axhausen. 2015. An integrated Bayesian approach for passenger flow assignment in metro networks. *Transportation Research Part C: Emerging Technologies* 52 (2015), 116–131.

[20] Yuxing Sun, Biao Leng, and Wei Guan. 2015. A novel wavelet-SVM short-time passenger flow prediction in Beijing subway system. *Neurocomputing* 166 (2015), 109–121.

[21] Evelien Van Der Hurk, Leo Kroon, Gábor Maróti, and Peter Vervest. 2015. Deduction of passengers' route choices from smart card data. *IEEE Transactions on Intelligent Transportation Systems* 16, 1 (2015), 430–440.

[22] Fei Wang, Chenhao Tan, Ping Li, and Arnd Christian König. 2011. Efficient document clustering via online nonnegative matrix factorizations. In *Proceedings of the 2011 SIAM International Conference on Data Mining*. SIAM, 908–919.

[23] Yu Wei and Mu-Chen Chen. 2012. Forecasting the short-term metro passenger flow with empirical mode decomposition and neural networks. *Transportation Research Part C: Emerging Technologies* 21, 1 (2012), 148–162.

[24] Billy M Williams and Lester A Hoel. 2003. Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results. *Journal of transportation engineering* 129, 6 (2003), 664–672.

[25] Huaxiu Yao, Xianfeng Tang, Hua Wei, Guanjie Zheng, Yanwei Yu, and Zhenhui Li. 2018. Modeling Spatial-Temporal Dynamics for Traffic Prediction. *arXiv preprint arXiv:1803.01254* (2018).

[26] Huaxiu Yao, Fei Wu, Jintao Ke, Xianfeng Tang, Yitian Jia, Siyu Lu, Pinghua Gong, and Jieping Ye. 2018. Deep multi-view spatial-temporal network for taxi demand prediction. *arXiv preprint arXiv:1802.08714* (2018).

[27] Xiuwen Yi, Yu Zheng, Junbo Zhang, and Tianrui Li. 2016. ST-MVL: filling missing values in geo-sensory time series data. (2016).

[28] Yu Zhang and Dit-Yan Yeung. 2012. Overlapping community detection via bounded nonnegative matrix tri-factorization. In *Proceedings of the 18th ACM SIGKDD*. ACM, 606–614.

[29] Juanjuan Zhao, Qiang Qu, Fan Zhang, Chengzhong Xu, and Siyuan Liu. 2017. Spatio-Temporal Analysis of Passenger Travel Patterns in Massive Smart Card Data. *IEEE Transactions on Intelligent Transportation Systems* (2017).

[30] Yu Zheng, Licia Capra, Ouri Wolfson, and Hai Yang. 2014. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 3 (2014), 38.

[31] Jingbo Zhou and Anthony KH Tung. 2015. Smiler: A semi-lazy time series prediction system for sensors. In *2015 ACM SIGMOD*. 1871–1886.