Predicting Mobile Users Traffic and Access-Time Behavior Using Recurrent Neural Networks

Abdulrahman Alamoudi*, Mingliu Liu**, Ali Payani*, Faramarz Fekri*, Deshi Li**
*School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, USA
{aalamoudi31, apayani, faramarz.fekri}@gatech.edu
**Electronic Information School, Wuhan University, Wuhan, China
{liumingliu, dsli}@whu.edu.cn

2021 IEEE Wireless Communications and Networking Conference (WCNC) | 978-1-7281-9505-6/20/$31.00 ©2021 IEEE | DOI: 10.1109/WCNC49053.2021.9417361

***Abstract*—Predicting mobile users' web-access behavior can have substantial impacts on resource allocation and cost reduction for wireless networks. Therefore, we propose a machine learning platform to forecast the web traffic and access time of mobile users. Based on the observation, the traffic patterns exhibit complex dependency on time, location, and popularity of webpages. Thus, a Recurrent Neural Network (RNN) with Long Short Term Memory (LSTM) is developed based on distinct engineered features to learn and predict users' web browsing activities. Then, to forecast the future access-time of mobile users, we propose a Self-exciting Memory Neural Network (SMNN). The access activities are modeled as self-exciting point processes, and intensities are adopted for prediction. Moreover, we extend the proposed predicting framework to cell towers. To cope with the diversity of traffic at the cell tower, we resort to clustering methods to group the similar users of each tower. Then, we develop an LSTM model for each cluster separately to predict the web domain traffic activities for the cell tower. Finally, we show that our proposed models outperform the baseline prediction models based on cellular networks dataset. We also show that for the cell tower access prediction task, the clustering method can significantly improve the prediction accuracy.**

***Index Terms*—Mobile Access Behavior, Mobile Behavior Prediction, LSTM, Time Prediction**

I. Introduction

Network operators and mobile users are continually struggling to reduce their network operation costs and data costs. Even though an average user consumes just under 2 GB of data per month from a smartphone today, this usage is projected to grow to approximately 9 GB per month within the next four years [1]. Given that, content can be obtained in a more effective way if it was prefetched ahead of time. For example, a smartphone user could be connected to faster/cheaper WiFi networks ($0.06/GB) for certain times of a day, while connected to slower/expensive cellular networks ($10/GB) for the other times. Thus, it is valuable to predict and fetch the mobile user's content during the cheaper connectivity for a future time when the user is likely to be connected to expensive networks. Hence, the content is not stale by the time the user needs it in a cost-saving manner. However, the prediction is quite challenging not only because of the content dynamics, but also the usage that is likely to occur much later should be foreseen. In most of the past works [2]–[4], behavior data from all users is collected at a central server and it is used to gain some insights about the users' activity patterns in various applications. However, such approaches have significant limitations on mobile devices as they can hardly be aware of other user's behavior. Further, for privacy reasons, users are often unwilling to share their information with a third-party. Thus, we propose a machine learning approach based on Recurrent Neural networks (RNNs) which can effectively combine the statistical dependencies and sequential/temporal behavior of a mobile user to learn and predict his/her web content. This is the departure from most existing works which rely on Markov chain and sequential patterns rule models [5]–[9].

Ideally, the user's content should be offloaded to WiFi in a time-shifted manner to obtain the most up-to-date content. To address this issue, it is essential to estimate the access time of mobile users. Learning models for the location and the mobility of the users have been widely discussed [10], [11], where most of them focused on predicting the user's movements via GPS data and mobility patterns. However, the prediction of access time is more concerned in this paper. Considering the advantages of self-exciting point process in modeling event rate [12], [13], we propose to predict future events based on the learning of dynamic correlations among user's web access-time.

Further, wireless network operators are continually exploring different strategies for optimizing resource allocation, including the spectrum and bandwidth. To alleviate the traffic burden on the cellular network during the peak period and provide better network connectivity, users' browsing patterns on cellular networks should be predicted. Then, some of the learned traffic load can be offloaded to WiFi Access Points (APs) in a time-shifted manner. However, the large volume of users and various access behaviors make the prediction quite complicated. To overcome this challenge, we propose expanding our work to predict group users' behavior in cell towers, where users with similar behavior would be clustered together. In this paper, we develop a machine learning prediction platform for mobile users, and the contributions are summarized as follows:

- We formulate the web access prediction problem for mobile users via Long Short Term Memory (LSTM) and

This work is supported in part by National Science Foundation under award ID MLWiNS-2003002 and a Gift from Intel Co.

devise the access features to elaborate on their behaviors.

- We model the access-time events of mobile users as a self-exciting process and propose Self-exciting Memory Neural Network (SMNN) to predict the access-time for future events.
- We extend our LSTM framework for forecasting the web-type traffic activities in cell towers by using clustering method.

II. User Web Browsing Behavior

A. Browsing Behavior Analysis

In this study, we use a dataset that contains mobile users' web access logs for a period of one month. This data is collected by a wireless provider from over 1000 cell tower locations. Each access log in the dataset contains the anonymized subscriber ID, session start time, session end time, cell tower ID, website ID, and traffic volume in kilobytes. Our data shows that for most mobile users, the content exhibits intricate dependency on features such as time of access, the location, and repeatability of web domains. For a random user, Fig. 1a shows the accessed websites with respect to the time of the day where we observe that different websites are accessed within different time stamps. Similarly, Fig. 1b shows the accessed website with respect to the user's location. We notice that different locations are associated with different contents. As such considering the current location of the user would be beneficial in our prediction model. Further, this data strongly suggests that there are highly repetitive patterns in the web access of the mobile users. We compute the probability that any user will be visiting a new unique website in the next access and we show the results in Fig. 1c. It indicates that approximately 40% of the users are likely to visit a new website with only 20% of the time. In other words, 80% of the websites' visits are repeated. Table I shows the access features devised for our model. For instance, to extract item 8 in the table, we used a Bayesian estimator with a uniform prior (Laplace estimator) for estimating the popularity of the target websites from the same user.

TABLE I: Features used in the prediction model

ID	Feature
1	previous accessed website $[1, N_{website}]$
2	location of the access $[1, N_{location}]$
3	duration of the access (min)
4	time of the day (min)
5	is weekend? $[T, F]$
6	time since last access for each website $[1, N_{website}]$ (min)
7	average time between consecutive accesses (min)
8	website popularity $[1, N_{website}]$ $[0, 1]$

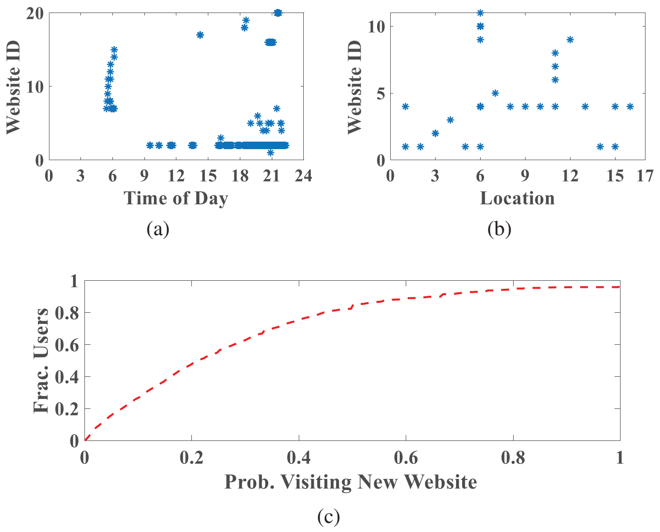


Fig. 1: Dataset analysis. (a) Absolute time of access; (b) Location of access; (c) Repeatability of websites visits

B. Browsing Behavior Prediction

In this work, we use LSTM [14] to predict a user's next webpage access given its activity history $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t\}$ where $\mathbf{x}_i$ is the feature vector at time i. Those features are used in our model to estimate the conditional probability distribution of the websites accesses $P(w^{t+1}|..., \mathbf{x}_{t-1}, \mathbf{x}_t)$ in the future time $t+1$. We assume that $\{w_1, w_2, ..., w_K\}$ are the list of K possible website choices for a user. Then, the website $w_j \in \{w_1, w_2, ..., w_K\}$ with the highest probability is predicted as the top choice for the next access. In order to ensure privacy and execute our model on mobile devices, we use a simpler gated design to increase the efficiency and reduce the computational cost. We aggregate all the hidden states using a fully connected output layer, thereby eliminating the need for an extra state and the output gate. In this framework, $\mathbf{z}_t$ encompasses the context $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t\}$ is used to quantify the conditional distribution of the next website access. In the following, we use lowercase bold letters to indicate vectors and uppercase bold letters for matrices. More formally, the proposed LSTM formulation is shown below:

$$
\begin{aligned}
\mathbf{h}_t^1 &= \phi^h(\mathbf{W}^{h_1 h_1}\mathbf{h}_{t-1}^1 + \mathbf{W}^{x h_1}\mathbf{x}_t + \mathbf{b}^{h_1}) \\
\mathbf{h}_t^i &= \phi^h(\mathbf{W}^{h_{i-1} h_i}\mathbf{h}_t^{i-1} + \mathbf{W}^{h_i h_i}\mathbf{h}_{t-1}^i + \mathbf{W}^{x h_i}\mathbf{x}_t + \mathbf{b}^{h_i}) \\
P(w^{t+1}|..., \mathbf{x}_{t-1}, \mathbf{x}_t) &= P(w^{t+1}|\mathbf{z}_t) \\
\mathbf{z}_t &= \sum_{l=1}^{L} \mathbf{W}^{h_l z}\mathbf{h}_t^l + \mathbf{b}^z, \qquad (1)
\end{aligned}
$$

where $i \in \{2, ..., L\}$, $\mathbf{W}$'s, $\mathbf{b}$'s, and L are weight matrices, bias vectors, and the number of layers, respectively. Further, ϕ^h is a non-linear activation function that is applied element-wise. We fix the weights of layers $\{\mathbf{h}^1, \mathbf{h}^2, ..., \mathbf{h}^L\}$ across the temporal dimension, but we use different weights for each layer. Based on this model, the probability distribution of the predicted website is calculated by applying softmax activation in the output layer as

$$
P(w^{t+1} = w_j|\mathbf{z}_t) = \frac{\exp(z_t^{w_j})}{\sum_{i=1}^{K}\exp(z_t^{w_i})}, j \in \{1, ..., K\}, \qquad (2)
$$

in order to decrease the distance between the estimated distribution and the true distribution we choose the negative log of likelihood as the cost function in (3). Then, we use Adaptive Moment Estimation (ADAM) [15] with a learning rate of 10^{-3} as an optimization algorithm that best fits our model .

$$L = -\sum_{t=0}^{N-1}\sum_{i=1}^{K} w_i^{t+1} log(\hat{w}_i^{t+1})$$
$$\hat{w}^{t+1} = P(w^{t+1}|\mathbf{z}_t). \tag{3}$$

C. Prediction within Time Interval

We further extend our model to predict a user's webpage access within a specified time interval given its past activity history. Specifically, at present time t, we wish to predict which website the user will access within time interval $(t, t+\Delta t]$. Since a mobile user's traffic behavior dynamically change over time, we want to keep track of the user's behavior and precisely predict the most probable webpage. This model estimates the conditional probability distribution of the websites including null activity (i.e. null activity indicates that a user has no browsing activity) from now to the end of the specific time interval. We assume that the set $(w_0, w_1, ..., w_{K-1})$ is the list of K possible website choices for a user where w_0 indicates no activity in the time interval. We design our model to predict the next domain within $\Delta t =$ an hour and $\Delta t =$ half an hour.

III. User Access-Time Prediction

To predict the access-time for a mobile user, we propose SMNN in this paper. In particular, the access activities are modeled as a self-exciting process. Hence, the dynamic of access events is learned based on intensity rather than timestamps. In SMNN, the neural network structure aims to quantify the impact of past events on future intensities, while the self-exciting model focuses on revealing the dynamic correlation and predicting the future occurring access-time.

A. Analysis of User Access-Time Sequences

User's time of access records can be regarded as a series of "randomly" occurring events, i.e., the so-called point process. However, access-time of the user's browsing patterns may cause bursty dynamics into the process. As shown in Fig. 2, the access event pattern of a random user is clustered stronger than that of a Poisson process. The phenomenon suggests that a non-homogeneous Poisson process is more applicable to model the rate of access events. Since the user's Internet access activities are modeled as a self-exciting point process, the impact can be specified via the intensity of future events, i.e., $\lambda(t)$, and can be expressed as

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{\mathbb{E}(N(t+\Delta t) - N(t)|\mathcal{H}_t)}{\Delta t}, \tag{4}$$

where $N(t)$ is the count number of events that happened till time t, and history $\mathcal{H}_t$ refers to the access-time history of the user, namely, $(t_1, t_2, ..., t_n)$.

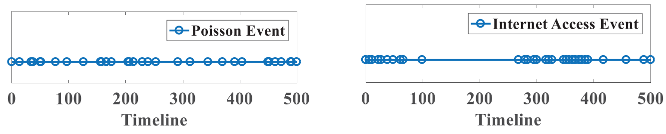


Fig. 2: Realisation of two point process. (a) An illustrated Poisson process; (b) An example of a user's access-time series

In order to specify the formulation of the intensity function, two characteristics are considered. First, the influence of past events on future intensity decreases with time. Fig. 3 provides an illustration based on Hawks process, which assumes the same excitation parameters for each past event. The comparison with statistical intensity calculated via real access data showed that the assumption might be unrealistic. Therefore, each past event's exciting pattern might differ, and it is considered the second feature, i.e., all past events would have an impact on the occurrence of the next event. Accordingly, the excitation of history intensities are assumed to be exponentially decaying with time, and the formulation is expressed as

$$\lambda(t) = \mu + \sum_{i=1}^{n} \alpha_i \exp(-\beta_i(t-t_i)), t \in [t_n, t_{n+1}), \tag{5}$$

where α_i is the initial excitation of event i, β_i is the decay rate of that excitation, and t_i is the occurring time of the i_{th} access event in the history.

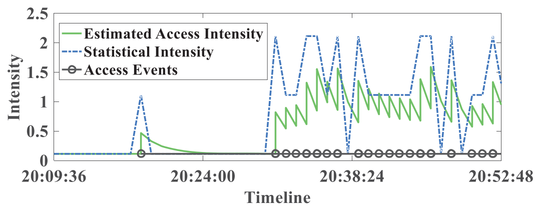


Fig. 3: Intensity illustrations of a user's access activities

B. Prediction Model of User Access-Time

Due to the advantage of capturing dynamic correlation, LSTM network is adopted to obtain the intensity parameters in (5). Thus, an accurate intensity model for event prediction is formulated as the SMNN model. As shown in Fig. 4, the input of the network is the time sequence of the mobile user's accessing history, and the space for output vector is $\mathbb{R}^{3\times n}$, which corresponds to the three parameters of the intensity model in (5). In particular, n is the length of the input time series, which is also called time steps. The baseline μ in (5) is the average of $(\mu_1, \mu_2, ...\mu_n)$. Then, the probability of one event happening during $[t_i, t_{i+1})$ can be expressed as

$$P\{N(t_{i+1}) - N(t_i) = 1|\mathcal{H}_t\} = -\Lambda(t)\exp(-\Lambda(t)), \tag{6}$$

where $\Lambda(t) = \int_{t_k}^{t} \lambda(t)dt$. In particular, we have

$$\int_{t_k}^{t} \lambda(t)dt = \mu t - \sum_{t_i < t_k} \frac{\alpha_i}{\beta_i} \exp(-\beta_i(t-t_i))|_{t_k}^{t}. \tag{7}$$

Therefore, the objective of training SMNN is to maximize the probability that events occurred as the input time series. Given by

$$L_{SMNN} = \sum_{i=1}^{N} \log(P\{N(t_{i+1}) - N(t_i) = 1|\mathcal{H}_t\}). \quad (8)$$

Then, combining (6) and (7), L_{SMNN} can be expressed by the output parameters. In our model, we attempt to explore the event occurring time based on the probability function. Similar to (6), supposing t is the last event time, then the probability of k events happened during $[t_i, t_{i+1})$ is given by

$$P\{N(t_{i+1}) - N(t_i) = k|\mathcal{H}_t\} = \frac{(-\Lambda(t))^k \exp(-\Lambda(t))}{k!}. \quad (9)$$

Denoting $P\{N(t_{i+1})-N(t_i) = k|\mathcal{H}_t\}$ as $\mathcal{P}_k(t)$. To predict the occurring time of the next event, we set $k = 1$, thus the time should be the minimum t that satisfies $\mathcal{P}_1(t) > \mathcal{P}_0(t)$. The prediction is based on the assumption that for a timestamp $t > t_l$, where t_l is the time of the most recent access event, if the probability of one event occurred before t is bigger than nothing happened, t would be provided as a reference for the occurring time of the next event. In particular, with k is set as different values, the prediction method can provide references for the occurring time of the next k_{th} event.

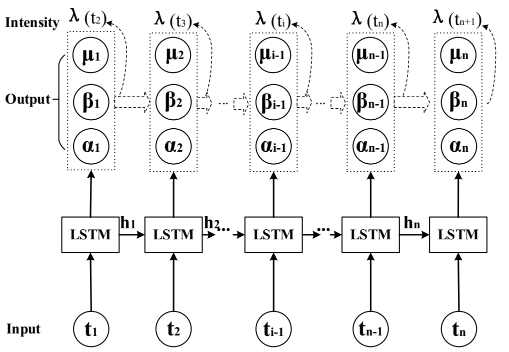


Fig. 4: The architecture of SMNN

IV. Prediction Model in Cell Tower

Next, our goal is to learn and predict the next webpage access by users from that cell tower. Building a single model for a cell tower using all mobile users can be problematic, as explained in the following: First, note that the web access activities in the cell tower are the interleaved traffic activities of mobile users; potentially causing confusion in the learning process. Second, different users could have similar traffic accessing and mobility behaviors. Therefore, the time-series of the aggregate traffic pattern from all users of a cell tower could introduce significant challenges to the machine learning model. As such, we propose to group the users in a cell tower based on their similarity in the webpage traffic activities into several clusters. In order to build these clusters, we model the distribution of each user's web access using a multinomial distribution in a specific cell tower. We then use Helinger distance as a measure of similarity between two probability distributions. Let $\mathbf{U} = [\mathbf{u_1}, \cdots, \mathbf{u_M}]$ be the set of distributions for M mobile users in a cell tower. Let $\mathbf{u_i} = \left[p_i^1, \cdots, p_i^K\right]$ be the i^{th} user distribution where p_i^l is the estimated probability for website l in a set of K websites. Then, we define a similarity index between user $\mathbf{u_i}$ and user $\mathbf{u_m}$ as

$$s_{im} = \frac{1}{\sqrt{2}} \sqrt{\sum_{j=1}^{K} (\sqrt{p_i^j} - \sqrt{q_m^j})^2}. \quad (10)$$

In this model, we cluster the users into separate super-user groups using K-mean, and we train the predictive LSTM model for each cluster separately. We use the same set of features as in Table I except the location feature due to irrelevancy. Further, in order to enhance the prediction accuracy, we train the models without interleaving the data from different users, as shown in Fig. 5. In the test phase, we time order the activities from different users. Since we do not know which user will generate the cell tower traffic, we take the average of the outcome of all models and declare the most likely one as the next traffic activity. Recall that the model is trained by minimizing the negative log-likelihood as in (3) using ADAM and learning rate of 10^{-3}.

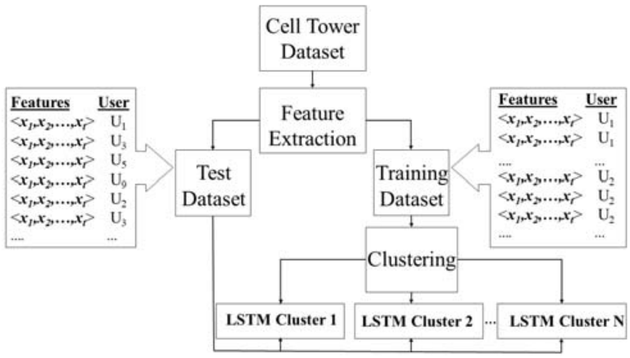


Fig. 5: Cell tower model diagram

V. Evaluation

A. User Behavior Prediction at Mobile Devices

We evaluated our model on access logs of more than 5000 real-world mobile users individually. For better modeling the data, we applied feature selection to study the effect of different features on the resulting model. Table II shows that feeding our model with all features performs the best average prediction accuracy among all users. It is noted that when the previously accessed website feature is eliminated, the accuracy of our predictive model is decreased significantly. This implies that the previously accessed website has a notable influence on our model. Further, Table III shows the structural design and hyperparameters selection of our model. To demonstrate the sequential nature of our traffic data, we compared the performance of our proposed LSTM model with Multiple Additive Regression Tree (MART) [4] and Multilayer Perceptron (MLP) [16] as benchmarks for top 1, top 3, and top

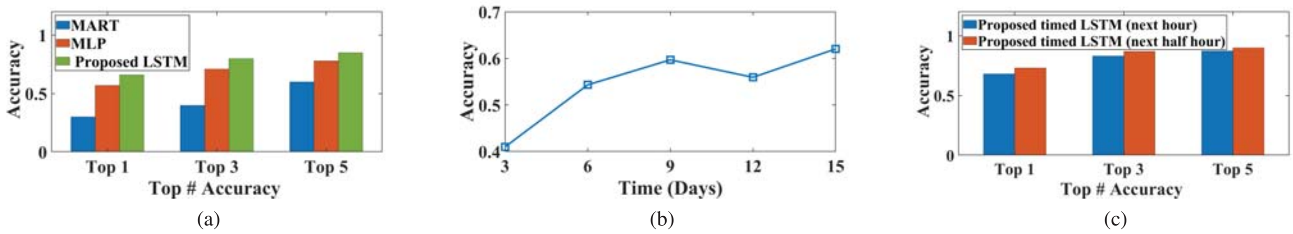


Fig. 6: Performance comparison of various user web-access modeling methods. (a) Proposed LSTM versus MART and MLP; (b) Proposed LSTM performance over time; (c) Proposed LSTM performance for both the next hour and next half hour.

5 most probable candidates. Fig. 6a summarizes the average prediction accuracy of each model. The results show the superiority of our model where the accuracy was 66% in comparison to 53% in MLP and 30% in MART. In another experiment, we evaluated the prediction accuracy as a function of time. For each user, we split the data for 15 days into five segments. Then for each segment, we evaluated the prediction accuracy when the model was trained on all past data up to that point. The results depicted in Fig. 6b suggest that the model converges to optimal performance quite fast on average while offering a reasonable accuracy after only one week of training data. Then, we evaluated the accuracy of the model for the prediction over a time interval Δt of half an hour and an hour, as shown in Fig. 6c. The results demonstrated that our model can predict the next webpage with an accuracy of 68% and 73% in the next hour and half an hour, respectively. This accuracy increases to 83% and 87% in the next hour and half an hour when considering the top 3 most probable candidates.

B. User Access-Time Prediction at Mobile Devices

We evaluated the performance of SMNN on 30 users who have the most access records and compared our proposed model to the Hawks process and LSTM as the baselines. All timestamps were transferred as absolute minutes and scaled into the range $[0, 1]$ in the network models. In this platform, root mean square error (RMSE) metric was adopted to measure the prediction accuracy. We tested the prediction performance for the next three future events, respectively. As shown in Fig. 7, although RMSE of Hawks process is decreased with the increase of past event length, the performance is inferior in comparison to LSTM and SMNN. Also, SMNN is more stable and better than the other two models in predicting the first next event. However, RMSE of our proposed method is dramatically increased when predicting the other two future events. The major reason lies in the incomplete description of the dynamic pattern of intensity, as decaying parameters of some closer future events have been ignored. Nevertheless, the results show that with a proper length of input events, SMNN achieves the best prediction performance among all three methods.

C. Web Access Prediction at the Cell Tower

We also evaluated our proposed LSTM model in the cell tower. First, we examined the feature selection and evaluated the sensitivity to different sets of features on our model, as shown in Table II. We observed that "time since last access for each website" negatively impacts our model's performance. Thus, we fed our models with all the features except "time since last access for each website". Moreover, Table III shows the structural parameters that we used in our model for the cell tower. Then, we implemented our model for a different number of clusters to obtain the best predictive result, as shown in Fig. 8a. We noticed that our model's accuracy, when implemented on more than two clusters, is diminished dramatically. This is most likely because of the insufficient dataset of users we had for the training. Our model is assessed on each cluster separately and then is reported over the average prediction accuracy. We evaluated our model on cell tower without clustering as a benchmark for Top n most probable websites for cell tower with the model using two clusters. Fig. 8b shows that clustering significantly increases the performance of our model.

VI. Conclusion

Understanding and utilizing users' behavior in web browsing is a pivotal element to mitigate the cost on users and network providers. Also, it would be crucial indication for future data caching design. The major contribution of this

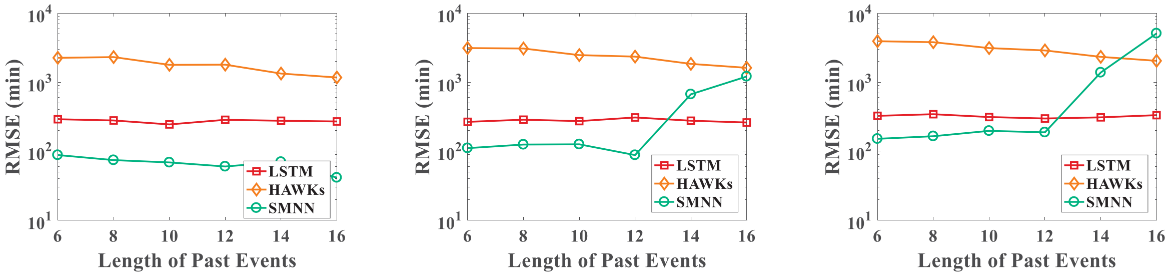
TABLE II: Feature sensitivity in user and cell tower models

Feature that is eliminated[a]	Accuracy (%)	
	User	Cell Tower
none	64	21
previous accessed website	54	3
location of the access	62	N/A
duration of the access	61	16
time of the day	61	10
is weekend	63	15
time since last access for each website	64	35
average time between consecutive accesses	60	17
website popularity	62	16

[a]Feature that is dropped in the evaluation process.

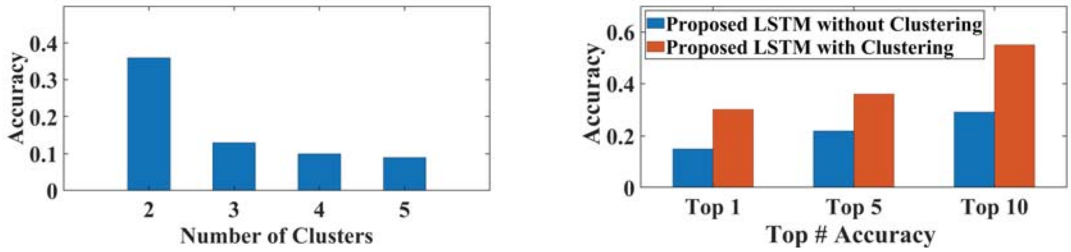
TABLE III: Hyperparameters in user and cell tower models

Parameter	User model	Cell Tower model
number of hidden layers	2	2
number of hidden units	100	1000
batch size	100	200
epochs	200	400
initial learning rate (α)	10^{-3}	10^{-3}
exponential decay rates $[\beta_1, \beta_2]$	$[0.9, 0.99]$	$[0.9, 0.99]$
denominator offset (ϵ)	10^{-8}	10^{-8}



(a) (b) (c)

Fig. 7: Performance comparison of predicting access-time. (a) the next 1*st* access-time; (b) the next 2*nd* access-time; (c) the next 3*rd* access-time



(a) (b)

Fig. 8: Cell tower behavior model result. (a) Proposed LSTM with clustering performance for different number of clusters; (b) Proposed LSTM with clustering performance in comparison with Proposed LSTM without clustering

paper is to enhance the prediction of the traffic in wireless networks by fully exploiting and engineering the sequential dependencies of the features. Specifically, we introduced three prediction algorithms, two for each mobile user and another for each cell tower. We analyzed detailed records of real mobile users and explored their browsing statistics in our model. Our evaluation results suggest that, when averaged over all users, we can accurately predict the next webpage to be accessed in 66% of the times. In particular, the predicted domain accuracy within a time window is 73%. In the access-time framework, the proposed SMNN model was shown to provide accurate time prediction for several future events and outperformed some state-of-the-art prediction methods in most cases. In the cell tower setup, the predictive model based on clustering showed an impressive performance improvement compared to the predictive model without clustering.

References

[1] By 2021, the average smartphone user will plow through 8.9 gb of data per month. [Online]. Available: http://bgr.com/2016/06/02/smartphone-data-usage-2021-gigabytes-erricson/

[2] V. N. Padmanabhan and J. C. Mogul, "Using predictive prefetching to improve world wide web latency," *ACM SIGCOMM Computer Communication Review*, vol. 26, no. 3, pp. 22–36, 1996.

[3] K. Lau and Y.-K. Ng, "A client-based web prefetching management system based on detection theory," in *International Workshop on Web Content Caching and Distribution*. Springer, 2004, pp. 129–143.

[4] D. Lymberopoulos, O. Riva, K. Strauss, A. Mittal, and A. Ntoulas, "Pocketweb: Instant web browsing for mobile devices," *SIGARCH Comput. Archit. News*, vol. 40, no. 1, pp. 1–12, Mar. 2012. [Online]. Available: http://doi.acm.org/10.1145/2189750.2150978

[5] H. Kawazu, F. Toriumi, M. Takano, K. Wada, and I. Fukuda, "Analytical method of web user behavior using hidden markov model," in *2016 IEEE International Conference on Big Data (Big Data)*. IEEE, 2016, pp. 2518–2524.

[6] M. Deshpande and G. Karypis, "Selective markov models for predicting web page accesses," *ACM Trans. Internet Technol.*, vol. 4, no. 2, pp. 163–184, May 2004. [Online]. Available: http://doi.acm.org/10.1145/990301.990304

[7] P. Fournier-Viger, T. Gueniche, and V. S. Tseng, "Using partially-ordered sequential rules to generate more accurate sequence prediction," in *International Conference on Advanced Data Mining and Applications*. Springer, 2012, pp. 431–442.

[8] J. Wei, Z. Shen, N. Sundaresan, and K.-L. Ma, "Visual cluster exploration of web clickstream data," in *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*. IEEE, 2012, pp. 3–12.

[9] D. Pai, A. Sharang, M. M. Yadagiri, and S. Agrawal, "Modelling visit similarity using click-stream data: A supervised approach," in *International Conference on Web Information Systems Engineering*. Springer, 2014, pp. 135–145.

[10] A. J. Nicholson and B. D. Noble, "Breadcrumbs: forecasting mobile connectivity," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, 2008, pp. 46–57.

[11] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting mobile 3g using wifi," in *Proceedings of the 8th international conference on Mobile systems, applications, and services*, 2010, pp. 209–222.

[12] H. Mei and J. M. Eisner, "The neural hawkes process: A neurally self-modulating multivariate point process," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 6754–6764.

[13] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec, "Seismic: A self-exciting point process model for predicting tweet popularity." New York, NY, USA: Association for Computing Machinery, 2015.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[16] M. W. Gardner and S. Dorling, "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences," *Atmospheric environment*, vol. 32, no. 14-15, pp. 2627–2636, 1998.