Dual Attention-Based Federated Learning for Wireless Traffic Prediction

***Abstract*—Wireless traffic prediction is essential for cellular networks to realize intelligent network operations, such as load-aware resource management and predictive control. Existing prediction approaches usually adopt centralized training architectures and require the transferring of huge amounts of traffic data, which may raise delay and privacy concerns for certain scenarios. In this work, we introduce a wireless traffic prediction method named *Dual Attention-Based Federated Learning* (FedDA), by which a high-quality prediction model is trained collaboratively by multiple edge clients. To simultaneously capture the various wireless traffic patterns and keep raw data locally, FedDA first groups the clients into different clusters by using a small augmentation dataset. Then, a quasi-global model is trained and shared among clients as prior knowledge, aiming to solve the statistical heterogeneity challenge confronted with federated learning. To construct the global model, a dual attention scheme is further proposed at the server by aggregating the intra- and inter-cluster models, instead of simply averaging the weights of local models. We verify FedDA on two real-world cellular datasets and experiment results indicate that FedDA outdoes state-of-the-art methods. The average mean squared error performance gain is up to 20%.**

***Index Terms*—wireless traffic prediction, federated optimization, spatio-temporal forecasting**

I. Introduction

Since the commercialization of fifth generation (5G) communication networks in 2019, the preliminary research on the potential features and enabling technologies for the sixth generation (6G) communications has attracted extensive attention in academia and industry [1, 2]. There are a set of emerging technologies and novel paradigms, e.g., terahertz spectrum, space-air-ground communications, large reflecting surfaces, and cognitive radios [3]. In addition, the communication research community has nevertheless reached a consensus that artificial intelligence (AI) is the key to implement novel paradigms, coordinate heterogeneous networks, organize various communication resources, and enable the truly smart 6G communications in the 2030s [4]. In particular, AI aided by big data and high-rate real-time transmission capability is expected to be the most efficient approach to reduce network overhead and elevate the quality of service (QoS) of both access and core networks [5].

To facilitate the fusion of AI and communication networks, wireless traffic prediction is indispensable. Wireless traffic prediction [6, 7] estimates traffic data volume in the future and provides the decision basis of communication network management and optimization [8]. With the predicted traffic data, proactive measures can be taken to mitigate the network congestion and outage caused by burst transmissions. Moreover, the heterogeneous service requirements, which are expected to become common in 6G communication networks [9], can be well satisfied with a lower cost by wireless traffic prediction. This will lead to a significant improvement in the QoS from both network's and user's perspectives.

Currently, most of wireless traffic prediction approaches are focusing on centralized learning scheme and involves transferring huge amount of raw data to a datacenter to learn a generalized prediction model. However, frequently transmission of training data and signaling overhead could easily exhaust the network capacity and yield negative impacts on payload transmissions. Thus new wireless traffic prediction approaches that can cope with the above challenges are needed.

The emergence and success of federated learning (FL) [10–14] make the prediction problem possible while keep data locally. In FL setting, lots of clients such as mobile phones and base stations (BSs) train a prediction model collaboratively. Only intermediate gradients or model parameters obtained by local training are transmitted to the central server rather than the raw data. There are enough reasons to support FL in the next-generation communications [15]. First, the advances of edge computing have paved the way for easily implementing FL in reality. As edge clients equip abundant computing resources, the centralized datacenter is not essential anymore and the delay of transferring raw data can be considerably reduced. In addition, FL makes data collection and model training extremely flexible and convenient. For instance, the clients can actively collect data during the day, and then collaboratively update the global model at the night, to enhance the prediction accuracy for the future usage.

Despite the promising application prospect, accurate wireless traffic prediction under the FL settings is still a major research challenge, especially the network-wide prediction. This is because user mobility can cause sophisticated spatio-temporal coupling among wireless traffic, which can hardly be captured and modeled. Furthermore, different BSs may have distinct traffic patterns which makes the traffic data highly heterogeneous and the learning and prediction on this kind of heterogeneous data is very challenging.

Therefore, to cope with the wireless traffic prediction issues for future communication networks, we propose a novel wireless traffic prediction framework named the dual attention-based federated learning (FedDA), by which a high-quality prediction model is trained collaboratively by multiple BSs. The FedDA framework relies on a set of state-of-the-art training paradigms, including a data augmentation assisted clustering strategy, an intermediate and auxiliary training

model, a dual attention-based model aggregation, and a hierarchical aggregation structure. Specifically, the processing of FedDA can be split into three stages to ensure high-accuracy, transferable, and secure training process for wireless traffic prediction.

We first introduce an augmentation assisted clustering strategy to group all BSs, i.e., clients in the context of FL, into a number of clusters depending on their augmented traffic patterns and geographic locations. Then, leveraging the augmented data collected from distributed BSs, a quasi-global prediction model can be constructed at the central server. This quasi-global model is used to mitigate the generalization difficulty of the global model caused by the statistical heterogeneity among traffic patterns collected from different clusters. Finally, instead of simply averaging the model weights collected from local clients to yield the global model, a dual attention-based model aggregation mechanism and a hierarchical aggregation structure are adopted at the central server. By introducing the dual attention and hierarchical settings, an adequate equilibrium between generality and specialty can be achieved.

Following the proposed FedDA framework and the descriptions given above, we summarize the contributions of this work as follows:

- We propose a data augmentation-assisted iterative clustering strategy, which takes the augmented data and geographic locations of clients as clustering reference to simultaneously capture various traffic patterns of clients and protect data privacy.
- We introduce a quasi-global model, which is an intermediate and auxiliary tool to mitigate the generalization difficulty of the global model caused by the statistical heterogeneity among traffic patterns collected from different clients.
- We design the FedDA framework consisting of two advanced settings for aggregations, which are the dual attention-based model aggregation mechanism and the hierarchical aggregation structure. In this way, the central server can capture not only the cluster-specific data patterns but also ensure the transferability of the global model.
- We verify the superiority of the FedDA framework by testing on two real-world datasets and compare the experimental results with those generated by existing algorithms.

The rest of this paper is organized as follows. Section II gives related works on wireless traffic prediction and FL. Section III gives the system model and problem statement. In section IV, we introduce our proposed dual attention mechanism in detail, including the data augmentation-assisted client clustering, mathematical expression of dual attention, and the corresponding optimization techniques. All the experiments are presented and discussed in Section V. Finally, this paper is concluded in Section VI.

II. Related Work

As the present work is closely related to wireless traffic prediction and FL, we review the most related achievements and milestones of these two research topics in this section.

A. Wireless Traffic Prediction

Recently, wireless traffic prediction has received a lot of attention as many tasks in wireless communications require accurate traffic prediction capabilities. It can be in essence framed as a time series prediction problem and the methods to solve it can be roughly classified into three categories, i.e., simple methods, parametric methods, and non-parametric methods.

Historical average and naïve methods are representatives of the first category [16]. The former predicts all future values as the average of the historical data, while the latter takes the last observation as the future. This kind of prediction method involves no complex computations, and thus makes it quite simple and easy to implement. However, as simple methods fail to capture the hidden patterns of wireless traffic, their prediction performance is relatively poor.

For the second category, i.e., parametric methods, the wireless traffic is modeled and predicted based on tools from statistics and probability theory. The most classical method is AutoRegressive Integrated Moving Average (ARIMA) [17]. To characterize the self-similarity and bursty of wireless traffic, ARIMA and its variants were explored in [18, 19]. In a recent study, [20] first decomposed the wireless traffic into regularity and randomness components. Then the authors demonstrated that the regularity component can be predicted through the ARIMA model, but the prediction of random components is impossible. Besides the ARIMA model, the α-stable model [21], entropy theory [22], and covariance functions [23] were also explored to perform wireless traffic prediction.

As machine learning and AI techniques continue their fast evolving, the non-parametric methods have achieved state-of-the-art results for wireless traffic prediction. Particularly, recent years have witnessed an obvious trend in solving wireless traffic prediction problems based on deep learning [24]. In [6], A hybrid deep learning framework was designed on the basis of autoencoder and Long-Short Term Memory networks (LSTM) to simultaneously capture the spatiotemporal dependence among different cells. Aiming to perform prediction on multiple cells, the researcher also introduced a multi-task learning framework by using LSTM [25]. Besides, the city-scale wireless traffic predictions are also investigated in [7, 26], in which the authors introduced novel prediction frameworks by modeling spatiotemporal dependence over cross-domain datasets.

All aforementioned works mainly focus on wireless traffic prediction in the centralized way. Our proposed framework in this paper differs from the above works, and we are trying to solve the wireless traffic prediction problem by a distributed architecture and federated learning.

B. Federated Learning

FL provides a distributed training architecture that can be jointly applied with many machine learning algorithms, in particular deep neural networks. In FL, a global model can be obtained by aggregating local clients' models. To obtain the global model, [10] introduced an aggregation method called federated averaging (FedAvg). Research shows that if the data is independent and identically distributed (IID), FedAvg achieves similar performance compared with centralized learning. However, when the client data is non-IID, the performance of FedAvg degrades greatly. To solve this problem, [27] proposed a data-sharing strategy by creating a small shared dataset that each client can access. This strategy can solve the statistical heterogeneity challenge confronted with FL. In [28], the authors introduced FedProx, which can be viewed as a generalization and re-parameterization of FedAvg, to tackle heterogeneity in federated networks. Besides, [29] introduced an attentive federated aggregation scheme, called FedAtt, by considering unequal contributions from different clients to the global model. This scheme improves the generalization ability of the global model and has been successfully used to solve the language modeling problem. For a more detailed introduction on the development of FL, please refer to a recent survey [12].

Our work is inspired by the above research, but we mainly focus on wireless traffic prediction problem, which is different from the above works. Also, because the wireless traffic data is highly heterogeneous, we propose a novel FedDA framework to solve this statistical challenge.

III. Problem Formulation and Preliminaries

In this section we first provide the problem formulation and then give the implementation details of FL corresponding to the formulated problem.

A. Problem Formulation

Given K BSs, each BS have its own local wireless traffic data, denoted as $d^k = \{d_1^k, d_2^k, \cdots, d_Z^k\}$ with a total of Z time intervals. The wireless traffic prediction problem can be described as the prediction of future traffic volume based on the current and the previous traffic volumes. Suppose d_z^k to be the target traffic volume required to be predicted, then the wireless traffic prediction problem can be described as

$$\hat{d}_z^k = f(d_{z-1}^k, d_{z-2}^k, \cdots, d_1^k; w), \tag{1}$$

where $f(\cdot)$ denote the chosen prediction model and w the corresponding parameters. The prediction model $f(\cdot)$ can be either in a linear form like linear regression or in a nonlinear form like deep neural networks.

For machine learning based wireless traffic prediction techniques, it is common to use only part of the historical traffic data as input features to reduce complexity. Thus, based upon d^k, a set of input-output pairs $\{x_i^k, y_i^k\}_{i=1}^n$ can be obtained by using sliding window scheme. x_i^k denotes the historical traffic data related to y_i^k, and we set it to $\{d_{z-1}^k, \cdots, d_{z-p}^k, d_{z-\phi 1}^k, \cdots, d_{z-\phi q}^k\}$. p and q are the sliding window sizes for capturing the closeness dependence and period dependence of wireless traffic data and ϕ is the periodicity [26, 30]. As we focus only on the one-step ahead prediction problem, so $y_i^k = d_z^k$. Thus, the problem formulated in (1) can be reformulated as

$$\hat{y}_i^k = f(x_i^k; w). \tag{2}$$

Our objective is to minimize the prediction error over all K BSs, thus the parameters w can be obtained by solving

$$\arg\min_w \left\{ \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}(f(x_i^k; w), y_i^k) \right\}, \tag{3}$$

where $\mathcal{L}$ is the loss function and the structure typically takes $|f(x_i^k; w) - y_i^k|^2$ or $|f(x_i^k; w) - y_i^k|$.

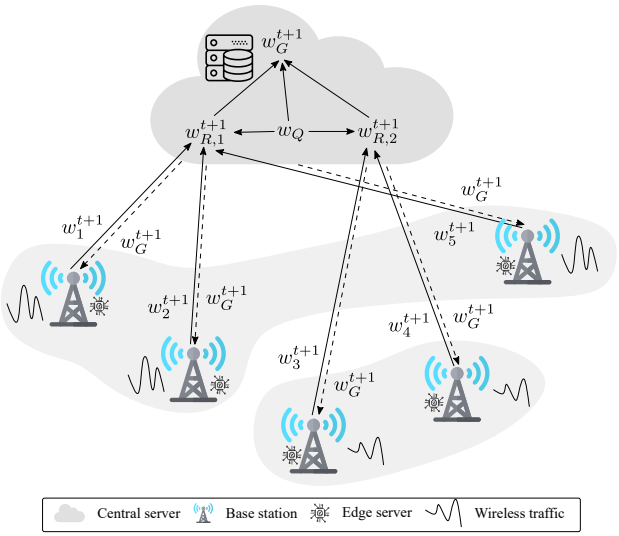
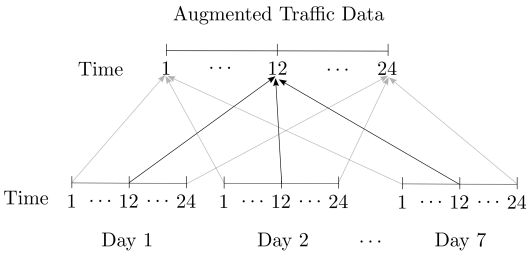


Fig. 1: System diagram of FedDA with five BSs in two clusters.

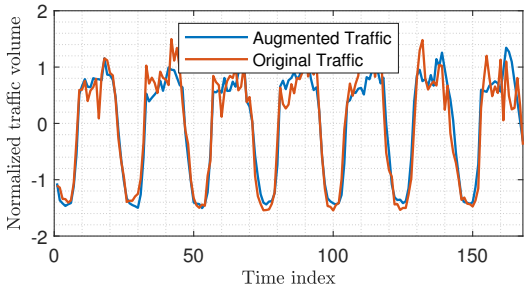
B. Preliminaries of Federated Learning

We try to solve (3) in a distributed manner, particularly under the cross-silo FL settings [14] and assume data are located at geo-distributed clients. After initializing the global model characterized by the parameters w, FL works as follows at the t-th training round:

1) The central server send the global model w^t to all BSs;
2) The BS treats the global model as prior knowledge and updates its local model based upon w^t with its local data. The local model update rule is given as follows: $w_k^{t+1} \leftarrow w_k^t - \eta \nabla_{w^t} \mathcal{L}(f(x^k; w^t), y^k)$, where η is learning rate and ∇_{w^t} is the gradient of loss function with respect to w^t;
3) The BS sends w_k^{t+1} to the central server;
4) The central server performs model aggregation (also known as federated optimization) based on local models. The most classical model aggregation scheme is the federated averaging [10], which can be written as $w^{t+1} \leftarrow \frac{1}{K} \sum_{k=1}^{K} w_k^{t+1}$.



(a) Wireless traffic augmentation strategy.



(b) Augmented versus original traffic data.

Fig. 2: Illustration on wireless traffic data augmentation and the comparison between the augmented and the original traffic data of a BS.

Algorithm 1: Iterative clustering strategy

Input: BS location $\{g_k\}_{k=1}^K$, augmented traffic $\{\tilde{d}^k\}_{k=1}^K$, cluster size C, cluster center $\{v_c\}_{c=1}^C$, and iteration threshold J.
Output: C clusters

```
Random initialize {v_c}_{c=1}^C
while j < J do
    Group {d̃^k}_{k=1}^K into C clusters (l_1, l_2, ⋯, l_C) by
      using K-Means with {v_c}_{c=1}^C;
    Group {g_k}_{k=1}^K into C clusters (l'_1, l'_2, ⋯, l'_C) by
      using K-Means
    if {l_c}_{c=1}^C is the same as {l'_c}_{c=1}^C then
        break
    Update cluster center {v_c}_{c=1}^C based on (l'_1, l'_2, ⋯,
      l'_C)
    j = j + 1
```

By running the above steps iteratively until the termination conditions are satisfied, a final global model w can be obtained.

IV. Proposed Framework

In this section, we present a detailed introduction of our proposed FedDA, and a demo FedDA system diagram with five BSs in two clusters is shown in Fig. 1. Specifically, FedDA consists of three steps:

1) Each BS performs the traffic augmentation procedure and sends the augmented data to the central server;
2) The central server yields a quasi-global model and clusters the BSs into different groups based on the augmented data and the geolocations of BSs;
3) The BSs cooperatively learn a global model under the organization of the central server by using the dual attention-based federated optimization.

In the following, we explain how to augment wireless traffic data and analyze the similarities between augmented data and original traffic data. After that, we introduce the iterative clustering strategy by taking into consideration both locations and traffic patterns of BSs, followed by a detailed elaboration of our proposed dual attention-based model aggregation scheme.

A. Wireless Traffic Data Augmentation

As urban areas have different functions to support the daily operation of a city, the traffic patterns of spatially distributed BSs differ a lot. Besides, users have different mobility and communication behaviors, which further enlarge the pattern diversity of wireless traffic. Therefore, wireless traffic data of different BSs has a high level of heterogeneity and is essentially non-IID. Performing FL over non-IID data is quite challenging as weight divergence exists when performing model aggregation on the server side [27]. Herein, we propose a augmentation-based data sharing strategy to conquer the statistical heterogeneity challenge.

Our augmentation strategy works as follows. The data is firstly divided into weekly slices according to the time index. Then for each week's traffic data, we compute the statistical average value for each time index and treat the obtained result as the augmented data. Finally, the augmented traffic is standardized to have zero mean and unit variance. An illustration of the augmentation procedure is displayed in Fig. 2a, and the comparison between the augmented data and the original data can be found in Fig. 2b.

From Fig. 2 we can observe that the proposed augmentation strategy is quite easy to implement and produce the augmented data, compared with traditional time series data augmentation strategies either in time domain or frequency domain [31]. Though the strategy is simple and straightforward, it works well and achieves a Pearson correlation coefficient of 0.9526, which indicates high similarities between augmented data and original data.

Each BS sends a very small part, say $\varphi\%$, of its augmented data to the central server. Note that compared with the size of the raw data, the augmented data size is much smaller. Based upon this augmentation dataset, a quasi-global model can be trained and treated as prior knowledge for all BSs. We use the term 'quasi-global' because the model is trained using augmented data of all BSs, instead of the original data. Even so, this model can still be used as prior knowledge of all BSs because of the high similarities between the augmented data and the original data. We characterize the quasi-global model by w_Q in Fig. 1, and it has exactly the same network architecture as the local models and the global model.

B. Iterative Clustering for BSs

As mentioned earlier, spatially distributed BSs have different traffic patterns. To capture the pattern heterogeneity among BSs and train an accurate prediction model suited for most BSs, we perform clustering analysis for BSs and propose an iterative clustering strategy to achieve this purpose by taking into consideration both the geo-locations and the traffic patterns of BSs. The detail is summarized in Algorithm 1.

For an arbitrary BS k, the central server knows its geolocation information g_k and stores its augmented traffic data $\tilde{d}^k$. By using a random initialization of C cluster centers $\{v_c\}_{c=1}^{C}$, we perform the K-Means algorithm on the augmented data $\{\tilde{d}^k\}_{k=1}^{K}$ and obtain the cluster labels of BSs (Line 3 of Algorithm 1). Then we use the location information $\{g_k\}_{k=1}^{K}$ as input and similarly perform the K-Means algorithm on it. This can yield C different clusters (Line 4 of Algorithm 1). If the clustering results on these two kinds of data are the same, then Algorithm 1 stops and returns the cluster label of each BS. If the yielded results are not the same, then based on the obtained cluster label on geo-location data, we compute the cluster center and use this to initialize the K-Means clustering on the traffic data. This indicates that the geo-location information is considered by the traffic pattern clustering process. The above steps are repeated on an iterative basis until the termination conditions are satisfied.

As shown in Fig. 1, the BSs are clustered into different clusters based on their geo-location and traffic pattern. After obtaining the cluster label of each BS, FedDA proceeds to the federated optimization, which will be introduced in the next subsection.

C. Dual Attention-Based Model Aggregation

One of the most fundamental part of FL is the model aggregation scheme, which involves constructing the final global model based on the received local ones. Herein, we design a novel federated optimization strategy, i.e., FedDA, for obtaining the global model. In particular, we introduce attention scheme into the model aggregation of FedDA and quantifies the contributions of both local models and the quasi-global model in a layer-wise manner. Fig. 3 shows an illustration of our proposed layer-wise dual attention-based model aggregation procedure. Note that we adopt a hierarchical learning scheme in FedDA. That is, there are two levels of model aggregation. The first one performs intra-cluster model aggregation, whose function is to obtain cluster models that capture the unique traffic patterns of each cluster. The second one performs inter-cluster model aggregation, after which the final global model that incorporates knowledge of all clusters is generated.

In Fig. 3, $w_{I,m}^{t+1}$ denotes the m-th input model, and there are M input models in total. w_O^{t+1} denotes the output model. For intra-cluster model aggregation in cluster c, $w_{I,m}^{t+1}$ belongs to a local model of that cluster, and w_O^{t+1} stands for $w_{R,c}^{t+1}$. For inter-cluster model aggregation, $w_{I,m}^{t+1}$ belongs to $\{w_{R,c}^{t+1}\}_{c=1}^{C}$, and the output is the global model w_G^{t+1}.

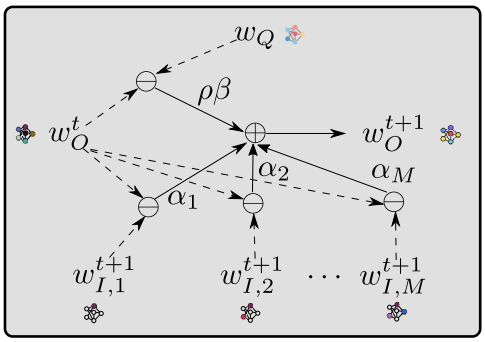


Fig. 3: Dual attention-based model aggregation.

The purpose of federated optimization on the central server side is to find an optimal global model that can have a strong generalization ability over all BSs. To achieve this, the global model should find a balance between capturing the unique and the common-shared traffic patterns of BSs. Thus, in our proposed scheme, we regard the optimization problem as finding a global model that is close to both of the local models and the quasi-global model in the parameter space, considering their different contributions during the model aggregation procedure. Consequently, the optimization objective is to minimize the sum-weighted distance among different models by using self-adaptive scores as weights. The federated optimization problem is formally defined as

$$\arg\min_{w_O^{t+1}} \left\{ \sum_{m=1}^{M} \frac{1}{2}\alpha_m \mathcal{L}(w_O^t, w_{I,m}^{t+1})^2 + \frac{1}{2}\rho\beta\mathcal{L}(w_O^t, w_Q)^2 \right\}, \tag{4}$$

where α_m and β represent attention weight vectors denoting the contributions of each layer of the m-th input model and the quasi-global model; ρ is a task-dependent regularization parameter and can be manually set based on experiment requirements.

To obtain the weights α_m, we use the attention mechanism and apply it to the layer-wise parameters. For the m-th input model, we denote the l-th layer's parameters as $w_{I,m}^l$. Similarly, we denote the l-th layer's parameters of the output model as w_O^l. The time stamps are omitted in $w_{I,m}^l$ and w_O^l for simplicity. Based on the layer-wise parameters, the distance between $w_{I,m}^l$ and w_O^l can be calculated by the Frobenius norm of their difference, which is expressed as

$$s_m^l = \mathcal{L}(w_{I,m}^l, w_O^l) = \|w_{I,m}^l - w_O^l\|_2^2. \tag{5}$$

Subsequently, the softmax function is applied to s_m^l and maps the non-normalized distance values to a probability distribution over the M input models. In this way, the contributions of these models can be determined. The standard softmax function $\sigma(\cdot)$ is described as

$$\alpha_m^l = \sigma(s_m^l) = \frac{e^{s_m^l}}{\sum_{m=1}^{M} e^{s_m^l}}. \tag{6}$$

Algorithm 2: Implementation of FedDA

Input: Wireless traffic data $\{x^k, y^k\}_{k=1}^K$; Quasi-global model, w_Q; Fraction of BSs, δ; Learning rate of local BS, η; Step size of server side, γ.

Output: Global model, w_G

```
for each round t = 1, 2, ··· , do
    m ← max(K · δ, 1)
    S_t ← a random set of m BSs
    for each client k ∈ S_t do
        w_k^{t+1} ← w_k^t − η∇_{w^t} L(f(x^k; w^t), y^k)
    for cluster c = 1, 2, ··· do
        S_c ← a set of BS with cluster label c
        Obtain w_{R,c}^{t+1} by using Equations (5) to (8)
    Obtain w_G^{t+1} by using Equations (5) to (8)
```

Similarly, the values of β can be obtained. After we get the α_m and β, the output model's parameters can be updated by using the gradient descent algorithm. We first compute the derivative of (4) in respect of w_O^t and obtain the corresponding gradient

$$\nabla = \sum_{m=1}^{m} \alpha_m (w_O^t - w_{I,m}^{t+1}) + \rho\beta(w_O^t - w_Q). \quad (7)$$

With the derived gradient, the output model parameters can be updated by

$$w_O^{t+1} = w_O^t - \gamma(\sum_{m=1}^{M} \alpha_m (w_O^t - w_{I,m}^{t+1}) + \rho\beta(w_O^t - w_Q)), \quad (8)$$

where γ is a predetermined step size that controls how much w_O should move in the direction of the opposite gradient in every iteration. The whole procedure of our proposed FedDA is summarized in Algorithm 2.

V. Experiments

In this section, we perform extensive experiments to verify the effectiveness and efficiency of FedDA. We begin with a brief introduction of the dataset, evaluation metrics, and baseline methods. Then, experimental settings, such as learning rate and batch size, are given. After that, we report the experimental results, including the overall prediction performance of various methods and the influence of learning (hyper-) parameters on prediction performance.

A. Dataset and Evaluation Metrics

The datasets used in this paper come from the *Big Data Challenge* [32] launched by Telecom Italia and mainly are Call Detail Records (CDR) of two Italian areas, i.e., the city of Milan and the province of Trentito. The area of Milan is divided into a grid of 10000 cells, and for Trentino, the grid is of 6575 cells. In each cell, the user's telecommunication activities are served and logged by the BS and thus we use BS or cell interchangeably to denote a cell. There are threes types of wireless traffic, which are corresponding to SMS, voice Call, and Internet services. The traffic is logged every ten minutes over two months, from $11/01/2013$ to $01/01/2014$. For experiments in the following subsections, the traffic is resampled into hourly to circumvent the data sparsity problem. These two datasets are publicly available and can be accessed on Harvard Dataverse [33, 34]. To evaluate prediction performance, two widely used regression metrics are adopted in this paper, i.e., mean squared error (MSE) and mean absolute error (MAE).

B. Baseline Methods

In this paper, we compare FedDA with the following five baseline methods.

- Lasso: A linear model for regression.
- Support Vector Regression (SVR) [35]: SVR is one of the most classical machine learning algorithms and has been successfully used for traffic prediction.
- LSTM [25]: LSTM has a strong ability to model time series dataset and normally has better prediction performance than linear models and shallow-learning models.
- FedAvg [10]: FedAvg is first proposed in the pioneering work of federated learning. It adopts an average of local weights for model aggregation.
- FedAtt [29]: This algorithm is similar to FedAvg. However, when performing model aggregation in the central server, it differentiates and quantifies the contributions of different client models to the global model.

The first three baselines are trained in a fully distributed way. That is, the model is trained per client. The latter two baselines and our FedDA are trained in a federated way.

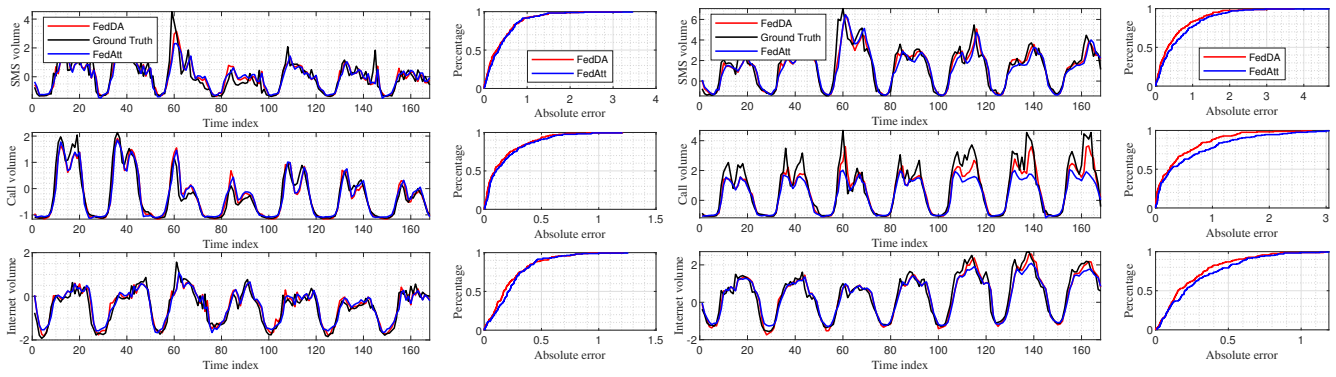
C. Experimental Settings and Overall Results

Without loss of generality, we randomly select 100 cells from each dataset and carry out experiments on the three kinds of wireless traffic of these cells. The traffic from the first seven weeks is used to train prediction models and the traffic from the last week is used for test. When constructing training samples using sliding window scheme, the length of closeness dependence p and periodicity dependence q are both set to 3. Considering that the edge client has limited computing power and thermal constraints, a relatively lightweight LSTM network is adopted. Specifically, the network has two LSTM layers and each layer has 64 hidden neurons, followed by a linear layer that maps the features to predictions[1]. All baselines except shallow learning algorithms share the same network architecture for the sake of fairness. Unless otherwise specified, we take 100 communication rounds between local clients and the central server and report results on the final model. The regularization term ρ is determined through a grid search with values ranging from -0.3 to 0.3 and step size 0.1. The cluster size C is set to 16. Similar to the standard settings in FL [10], the values of local epochs and local batch

[1]FedDA has a space complexity of $4\sum_{l=1}^{2} H_l(I_l + H_l + 2)$ and a time complexity of $4(p+q)B\sum_{l=1}^{2} H_l(I_l + H_l + 4)$, where l indexes the layers. I_l and H_l represent the input size and hidden size of the l-th layer and B is the batch size.

TABLE I: Prediction performance comparisons among different methods in terms of MSE and MAE on two datasets ('↑' denotes the performance gain of FedDA over FedAtt).

Methods	Milano						Trento					
	MSE			MAE			MSE			MAE		
	SMS	Call	Internet	SMS	Call	Internet	SMS	Call	Internet	SMS	Call	Internet
Lasso	0.7580	0.3003	0.4380	0.6231	0.4684	0.5475	4.7363	1.6277	5.9121	1.3182	0.8258	1.5391
SVR	0.4144	0.0919	*0.1036*	0.3528	0.1852	*0.2220*	5.2285	1.7919	5.9080	1.0390	0.5656	1.0470
LSTM	0.5608	0.1379	0.1697	0.4287	0.2458	0.2936	3.6947	1.1378	4.6976	0.9426	0.5013	1.1193
FedAvg	0.3744	0.0776	0.1096	0.3386	0.1838	0.2319	2.2287	1.6048	4.7988	0.7416	0.5319	1.0668
FedAtt	0.3667	0.0774	0.1096	0.3375	*0.1837*	0.2321	2.1558	1.5967	4.7645	0.7444	0.5306	1.0629
FedDA (φ=1)	0.3559	*0.0752*	0.1118	0.3353	0.1820	0.2367	2.1468	1.4925	4.4335	0.7478	0.5140	1.0212
FedDA (φ=10)	*0.3481*	0.0753	0.1062	*0.3321*	0.1810	0.2275	*2.0719*	*1.1699*	*3.9266*	*0.7320*	*0.4543*	*0.9504*
FedDA (φ=100)	**0.3322**	**0.0659**	**0.1033**	**0.3214**	**0.1741**	**0.2211**	**1.9703**	**1.0592**	**2.4473**	**0.6920**	**0.4281**	**0.7471**
↑ (φ=100)	+9.4%	+14.9%	+5.8%	+4.8%	+5.2%	+4.7%	+8.6%	+33.7%	+48.6%	+7.0%	+19.3%	+29.7%



(a) Results on the Milano dataset.

(b) Results on the Trento dataset.

Fig. 4: Comparisons between the predicted values and the ground truth values and the corresponding error analysis.

size are set to 1 and 20, respectively. In each communication round, 10% percent of the cells are involved in model training. Stochastic gradient descent is adopted to update our model with learning rate 0.01.

The experimental results of different prediction methods are presented in Table I. Note that in Table I, our proposed method have three variations on the basis of how many augmented data samples shared. In reality, the amount of transferred data samples can be flexibly adjusted according to network situation. From this table we can tell that our proposed method, FedDA, outperforms all the baseline methods for all kinds of wireless traffic in both datasets, even with only 1% of the augmentation data is shared. Specifically, for the SMS, CALL, and Internet traffic of the Milano dataset, compared with the best-performing method in baselines, namely FedAtt, FedDA[2] can offer MSE gains of 9.4%, 14.9%, and 5.8%, respectively. Likewise, for the Trento dataset, FedDA yields performance gains of 8.6% (SMS), 33.7% (CALL), and 48.6% (Internet), respectively. In terms of the metric of MAE, though the improvements are not as remarkable as MSE, it is still obvious that an average of 4.9% (18.7%) performance gain can be achieved for the Milano (Trento) dataset. We can also notice that the prediction performance of FedDA improves consistently with the increase of shared augmentation data size as the quasi-global model can capture the traffic patterns better when more data samples are available. The success of FedDA can be attributed to the following reasons:

- Compared with fully distributed algorithms (SVR and LSTM), which only consider temporal dependence of wireless traffic, FedDA can capture both the spatial dependence and the temporal dependence by means of model fusion, and is thus more robust;
- Compared with conventional FL algorithms (FedAvg and FedAtt), the introduced clustering strategy makes the learning process of FedDA case-specific, and the dual attention scheme greatly reduces data heterogeneity. Therefore, FedDA has a high generalization ability;
- FedDA can balance between capturing the unique characteristics of a cluster and the shared macro traffic patterns among different clusters, and thus has have more accurate predictions.

Besides, the results also indicate that in comparison with fully distributed algorithms, FL-based algorithms can achieve better predictions, and FedAtt achieves the second best predic-

[2]Here and in the next experiments, we give the results of FedDA with φ=100 unless otherwise specified.

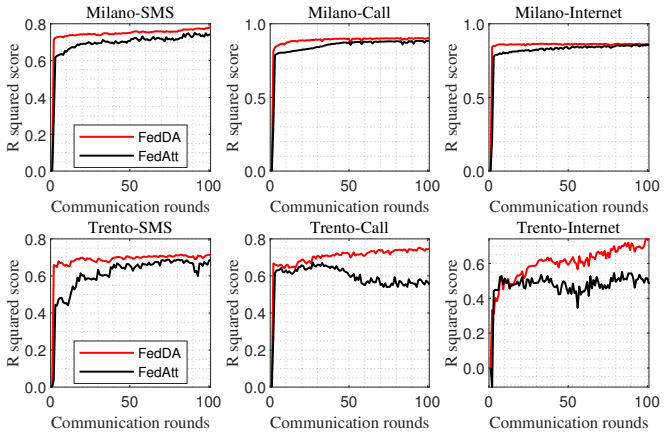


Fig. 5: Prediction accuracy versus communication rounds.

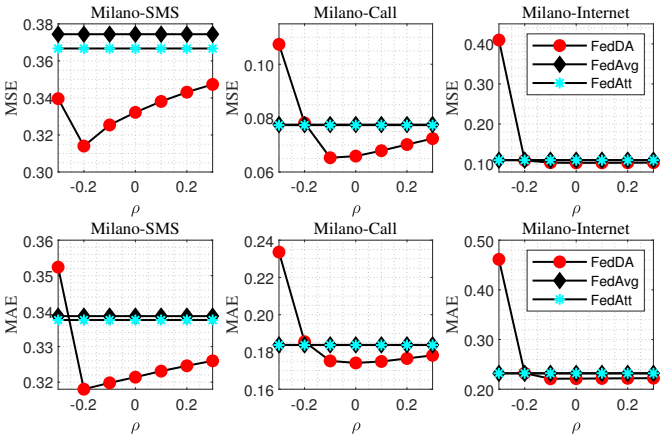


Fig. 6: Influence of ρ on prediction performance for the Milano dataset.

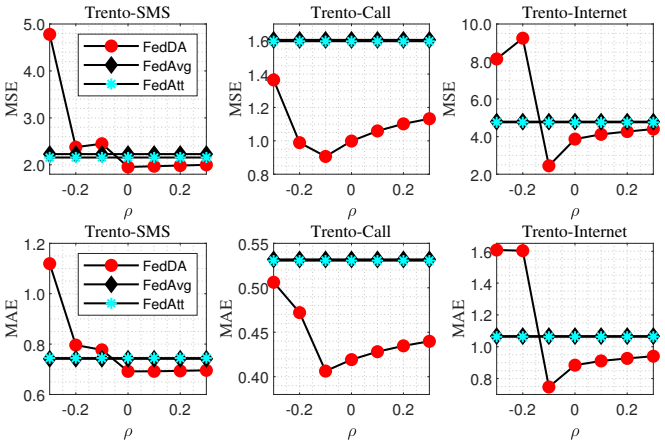


Fig. 7: Influence of ρ on prediction performance for the Trento dataset.

tion performance, followed by the classic FedAvg algorithm. This is rather intuitive, since there is no knowledge sharing when training prediction models with fully distributed algorithms. The lack of knowledge sharing results in a loss of prediction accuracy.

To further evaluate the predictive ability of different algorithms, the comparisons between the predicted and the ground truth values of different algorithms are given in Fig. 4. The results on cumulative distribution functions (CDFs) of absolute prediction error are also included in Fig. 4 for quantitatively measuring the goodness of prediction models. Fig. 4a (Fig. 4b) represents results on the Milano (Trento) dataset. More specifically, the left three subfigures of Fig. 4a (Fig. 4b) denote the comparisons between predictions and the ground truth for the SMS, Call, and Internet traffic of randomly selected cells, and the right three subfigures are the corresponding CDFs of errors. Here, we choose FedAtt as the benchmark for performance comparison, since it achieves the best performance among all baseline methods in Table I. By observing Fig. 4, we can tell that FedDA obtains consistent better prediction performance than FedAtt, on all three kinds of wireless traffic. Meanwhile, it has smaller prediction errors, especially when the traffic volume comes to high and unstable.

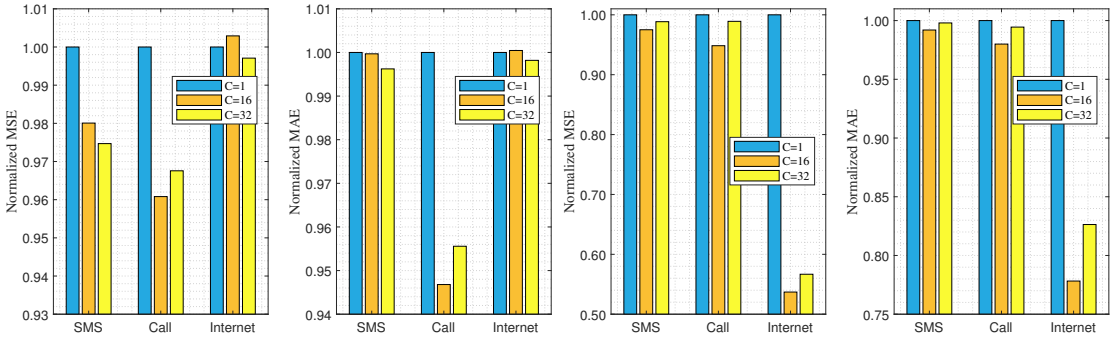
For prediction errors, taking the SMS traffic of the Trento dataset for example, there are approximately 83% errors that are smaller than 1 for FedDA, while the case for FedAtt is about 76%. Moreover, the average prediction errors for FedAtt and FedDA are 0.65 and 0.54, respectively. Based on the above evaluation, we can summarize that FedDA can achieve more accurate prediction results than those of baseline methods.

D. Communication rounds versus prediction accuracy

In FL or any other distributed learning frameworks, it is assumed that communication resources are more precious than computation resources and fewer communications are preferred. Thus, in this subsection, we report the prediction accuracy along with each communication round (epoch) and use R-squared score to denote the accuracy as it reflects how well ground truth values are predicted by the model [36]. The obtained results are summarized in Fig. 5, in which the upper (lower) three subfigures represent results on the Milano (Trento) dataset. From Fig. 5 we can observe that FedDA achieves higher accuracy on both two datasets and its advantages are clearer on the Trento dataset. More importantly, FedDA needs much fewer communications to achieve a certain prediction accuracy. Take the Milano dataset as an example, after 30 communication rounds, FedDA can achieve accuracies of 0.74, 0.89, and 0.86 for the SMS, Call, and Internet traffic, respectively. While for FedAtt, the achieved accuracies for the SMS, Call, and Internet traffic are 0.69, 0.84, and 0.83, respectively. Thus, here we argue that our proposed method is communication-efficient, which is key for performing learning tasks at the edge.

E. Effect of ρ on Prediction Performance

As ρ is a task-dependent regularizer and it impacts the prediction performance a lot, we present the prediction performance of FedDA along with ρ and summarize the results in Fig. 6 and Fig. 7 for the Milano and Trento datasets, respectively. In both figures, the value of ρ is ranging from



(a) Results on the Milano dataset. (b) Results on the Trento dataset.

Fig. 8: Cluster size versus prediction performance.

-0.3 to 0.3 with a step size of 0.1 and the obtained MSE (MAE) results are given in the upper (lower) three subfigures. Besides FedDA, the other two FL-based methods, i.e., FedAvg and FedAtt, are also included for comparison purpose but their MSE and MAE results keep constant as they are not affected by ρ. We can tell from Fig. 6 and Fig. 7 that for FedDA, with the increase of ρ, the values of MSE and MAE first decrease rapidly, and then slowly increase. For the other two baselines, FedAtt achieves slightly better results than FedAvg. Though ρ has a great influence on the prediction performance, FedDA generally can yield lower MSE and MAE values than FedAvg and FedAtt. Take the SMS traffic of the Milano dataset as an example, the obtained MSE results are always better than FedAvg and FedAtt regardless of the choice of ρ; while for the metric of MAE, similar conclusion hold except that ρ=-0.3, by which FedDA achieves worse results than FedAvg and FedAtt. Nonetheless, the optimal values of ρ can be determined by using a grid search strategy during model training and the cost is low. The results in these two figures demonstrate that the dual attention scheme in FedDA can indeed improve prediction performance by introducing prior knowledge and the influence varies on different datasets.

F. Effect of C on Prediction Performance

The cluster size C determines how many cells are involved in model aggregation and it also affects the final prediction performance. Thus, in this subsection, we explore how the cluster size affects the prediction performance of FedDA and the obtained MSE and MAE results are plotted in Fig. 8. In particular, Fig. 8a (Fig. 8b) shows the results on the Milano (Trento) dataset. We consider three scenarios, i.e., C=1, C=16, and C=32. Note that C=1 means no clustering adopted in prediction. In addition, to make the comparison clearer, the MSE and MAE results of C=16 and C=32 are normalized based on the results of C=1. We can observe that the choices of C yield different influences on the prediction performance of FedDA. In most cases, introducing the clustering strategy can indeed lead to lower prediction errors. Specifically, we can observe from Fig. 8a that FedDA achieves considerable performance improvements when cluster size is 16 or 32, for the SMS and Call traffic. For the Internet traffic, though the performance degrades slightly when C=16, it improves when C=32. For the Trento dataset, introducing the clustering strategy can always yield better prediction performance than not, especially for the Internet traffic, on which the improvement is up to 50%. Overall, the results in Fig. 8 demonstrate the superiority of introducing the clustering into FedDA. This is because the cluster size C controls the specialty of FedDA and thereby affects the global model. If no clustering strategy is involved, all data is mixed together to generate the global model. In this case, some unique traffic patterns hidden in the data cannot be captured by FedDA and hence leads to performance degradation.

VI. Conclusion

In this work, we investigated the wireless traffic prediction problem and proposed a novel framework called FedDA. To deal with the heterogeneity of wireless traffic data, we proposed a data-sharing strategy in FedDA by transferring a small augmented traffic dataset to the central server, by which a quasi-global model is obtained and shared among all BSs. Besides, we also introduced an iterative clustering algorithm to cluster BSs into different groups, by considering both the wireless traffic pattern and the geo-location information. To enhance the generalization ability of the global model, we proposed a dual attention-based model aggregation scheme by paying attention to the unequal contributions of different local models and the quasi-global model. The aggregation scheme is applied over a hierarchical architecture so as to capture both intra-cluster and inter-cluster patterns of wireless data traffic. Finally, we verified the effectiveness and efficiency of FedDA on two real-world datasets.

References

[1] K. David and H. Berndt. 6G vision and requirements: Is there any need for beyond 5G? *IEEE Vehicular Technology Magazine*, 13(3):72–80, 2018.

[2] B. Zong, C. Fan, X. Wang, X. Duan, B. Wang, and J. Wang. 6G technologies: Key drivers, core requirements, system architectures, and enabling technologies. *IEEE Vehicular Technology Magazine*, 14(3):18–27, 2019.

[3] S. Dang, O. Amin, B. Shihada, and M.-S. Alouini. What should 6G be? *Nature Electronics*, 3(1):20–29, 2020.

[4] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y. A. Zhang. The roadmap to 6G: AI empowered wireless networks. *IEEE Communications Magazine*, 57(8):84–90, 2019.

[5] M. Yao, M. Sohul, V. Marojevic, and J. H. Reed. Artificial intelligence defined 5G radio access networks. *IEEE Communications Magazine*, 57(3):14–20, 2019.

[6] J. Wang, J. Tang, Z. Xu, Y. Wang, G. Xue, X. Zhang, and D. Yang. Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, 2017.

[7] C. Zhang, H. Zhang, J. Qiao, D. Yuan, and M. Zhang. Deep transfer learning for intelligent cellular traffic prediction based on cross-domain big data. *IEEE Journal on Selected Areas in Communications*, 37(6):1389–1401, 2019.

[8] Y. Xu, F. Yin, W. Xu, J. Lin, and S. Cui. Wireless traffic prediction with scalable gaussian process: Framework, algorithms, and verification. *IEEE Journal on Selected Areas in Communications*, 37(6):1291–1306, 2019.

[9] N. Kato, B. Mao, F. Tang, Y. Kawamoto, and J. Liu. Ten challenges in advancing machine learning technologies toward 6G. *IEEE Wireless Communications*, 27(3):96–103, 2020.

[10] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pages 1273–1282, 2017.

[11] Q. Yang, Y. Liu, T. Chen, and Y. Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2):1–19, 2019.

[12] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith. Federated learning: Challenges, methods, and future directions. *arXiv preprint arXiv:1908.07873*, 2019.

[13] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konecny, S. Mazzocchi, H. B. McMahan, et al. Towards federated learning at scale: System design. *arXiv preprint arXiv:1902.01046*, 2019.

[14] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

[15] N. H. Tran, W. Bao, A. Zomaya, M. N. H. Nguyen, and C. S. Hong. Federated learning over wireless networks: Optimization model design and analysis. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pages 1387–1395, 2019.

[16] R. J. Hyndman and G. Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[17] J. M. Hamilton. *Time Series Analysis*, volume 2. Princeton New Jersey, 1994.

[18] Y. Shu, M. Yu, O. Yang, J. Liu, and H. Feng. Wireless traffic modeling and prediction using seasonal arima models. *IEICE Transactions on Communications*, 88(10):3992–3999, 2005.

[19] B. Zhou, D. He, and Z. Sun. Traffic predictability based on ARIMA/GARCH model. In *2006 2nd Conference on Next Generation Internet Design and Engineering*, pages 200–207, Apr. 2006.

[20] F. Xu, Y. Lin, J. Huang, D. Wu, H. Shi, J. Song, and Y. Li. Big data driven mobile traffic understanding and forecasting: A time series approach. *IEEE Transactions on Services Computing*, 9 (5):796–805, 2016.

[21] R. Li, Z. Zhao, J. Zheng, C. Mei, Y. Cai, and H. Zhang. The learning and prediction of application-level traffic data in cellular networks. *IEEE Transactions on Wireless Communications*, 16(6):3899–3912, 2017.

[22] R. Li, Z. Zhao, X. Zhou, J. Palicot, and H. Zhang. The prediction analysis of cellular radio access network traffic: From entropy theory to networking practice. *IEEE Communications Magazine*, 52(6):234–240, 2014.

[23] X. Chen, Y. Jin, S. Qiang, W. Hu, and K. Jiang. Analyzing and modeling spatio-temporal dependence of cellular traffic at city scale. In *2015 IEEE International Conference on Communications (ICC)*, pages 3585–3591, 2015.

[24] L. Nie, D. Jiang, S. Yu, and H. Song. Network traffic prediction based on deep belief network in wireless mesh backbone networks. In *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–5, 2017.

[25] C. Qiu, Y. Zhang, Z. Feng, P. Zhang, and S. Cui. Spatio-temporal wireless traffic prediction with recurrent neural network. *IEEE Wireless Communications Letters*, 7(4):554–557, 2018.

[26] C. Zhang, H. Zhang, D. Yuan, and M. Zhang. Citywide cellular traffic prediction based on densely connected convolutional neural networks. *IEEE Communications Letters*, 22(8):1656–1659, 2018.

[27] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.

[28] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems 2020*, 2020.

[29] S. Ji, S. Pan, G. Long, X. Li, J. Jiang, and Z. Huang. Learning private neural language modeling with attentive aggregation. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2019.

[30] J. Zhang, Y. Zheng, and D. Qi. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 1655–1661. AAAI Press, 2017.

[31] Q. Wen, L. Sun, X. Song, J. Gao, X. Wang, and H. Xu. Time series data augmentation for deep learning: A survey. *arXiv preprint arXiv:2002.12478*, 2020.

[32] G. Barlacchi, M. D. Nadai, R. Larcher, A. Casella, C. Chitic, G. Torrisi, F. Antonelli, A. Vespignani, A. Pentland, and B. Lepri. A multi-source dataset of urban life in the ity of Milan and the Province of Trentino. *Scientific Data*, 2:150055, 2015.

[33] Telecom Italia. Telecommunications - SMS, Call, Internet - TN, 2015. URL `https://doi.org/10.7910/DVN/QLCABU`.

[34] Telecom Italia. Telecommunications - SMS, Call, Internet - MI, 2015. URL `https://doi.org/10.7910/DVN/EGZHFV`.

[35] H. Feng, Y. Shu, S. Wang, and M. Ma. Svm-based models for predicting wlan traffic. In *2006 IEEE International Conference on Communications*, volume 2, pages 597–602, 2006.

[36] A. Cameron and F. A. G. Windmeijer. R-squared measures for count data regression models with applications to health-care utilization. *Journal of Business & Economic Statistics*, 14(2): 209–220, 1996.