Journal of Network and Computer Applications 121 (2018) 59–69

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Deep mobile traffic forecast and complementary base station clustering for C-RAN optimization

Longbiao Chen[a,b], Dingqi Yang[c], Daqing Zhang[d], Cheng Wang[a], Jonathan Li[a], Thi-Mai-Trang Nguyen[b,*]

[a] Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Information Science and Engineering, Xiamen University, China
[b] Sorbonne University, UMR 7606, LIP6, France
[c] eXascale Infolab, University of Fribourg, Switzerland
[d] Institut Mines-Télécom, Télécom SudParis, CNRS SAMOVAR, France

ARTICLE INFO

Keywords:
Deep learning
Mobile network
Big data analytics
C-RAN

ABSTRACT

The increasingly growing data traffic has posed great challenges for mobile operators to increase their data processing capacity, which incurs a significant energy consumption and deployment cost. With the emergence of the Cloud Radio Access Network (C-RAN) architecture, the data processing units can now be centralized in data centers and shared among base stations. By mapping a cluster of base stations with complementary traffic patterns to a data processing unit, the processing unit can be fully utilized in different periods of time, and the required capacity to be deployed is expected to be smaller than the sum of capacities of single base stations. However, since the traffic patterns of base stations are highly dynamic in different time and locations, it is challenging to foresee and characterize the traffic patterns in advance to make optimal clustering schemes. In this paper, we address these issues by proposing a deep-learning-based C-RAN optimization framework. First, we exploit a Multivariate Long Short-Term Memory (MuLSTM) model to learn the temporal dependency and spatial correlation among base station traffic patterns, and make accurate traffic forecast for a future period of time. Afterwards, we build a weighted graph to model the complementarity of base stations according to their traffic patterns, and propose a Distance-Constrained Complementarity-Aware (DCCA) algorithm to find optimal base station clustering schemes with the objectives of optimizing capacity utility and deployment cost. We evaluate the performance of our framework using data in two months from real-world mobile networks in Milan and Trentino, Italy. Results show that our method effectively increases the average capacity utility to 83.4% and 76.7%, and reduces the overall deployment cost to 48.4% and 51.7% of the traditional RAN architecture in the two datasets, respectively, which consistently outperforms the state-of-the-art baseline methods.

1. Introduction

Today, mobile network data traffic is growing explosively as Internet-enabled smartphones and tablets become increasingly popular (Zheng et al., 2016). According to Cisco (2016), global mobile network data traffic has grown 18-fold over the past five years, and the next-generation cellular systems (e.g., 5G) are expected to experience tremendous data traffic growth (Sigwele et al., 2017). In order to accommodate the fast growing data traffic demand, mobile network operators need to increase their *data processing capacity*, such as deploying more base stations, and adding more data processing units to base stations. Consequently, the *capital expenditures* of deploying these network infrastructures are becoming increasingly high, and may harm operator's revenue as network scale grows (J. Research, 2011). Moreover, the *operating expenses* of mobile network infrastructures, such as energy consumption and maintenance spending, are substantially increasing (Li et al., 2011). Therefore, optimizing the capital expenditures and operating expenses has become a necessity for mobile network operators (Checko et al., 2015; Gandotra and Jha, 2017).

Even though the overall data traffic demand of the mobile network is growing, the demand in different areas and during different periods of time is not evenly distributed (Chen et al., 2017a). For example, as shown in Fig. 1a, the traffic in a business district (denoted as a blue solid line) observes peaks during working hours, while the traffic in a residential area (denoted as a red dashed line) is relatively higher during evening hours than in working hours. Such a *spatial-temporal*

* Corresponding author. Sorbonne University, UMR 7606, LIP6, France.
E-mail addresses: longbiaochen@xmu.edu.cn (L. Chen), thi-mai-trang.nguyen@lip6.fr (T.-M.-T. Nguyen).

https://doi.org/10.1016/j.jnca.2018.07.015
Received 6 April 2018; Accepted 23 July 2018
Available online 31 July 2018
1084-8045/ © 2018 Elsevier Ltd. All rights reserved.

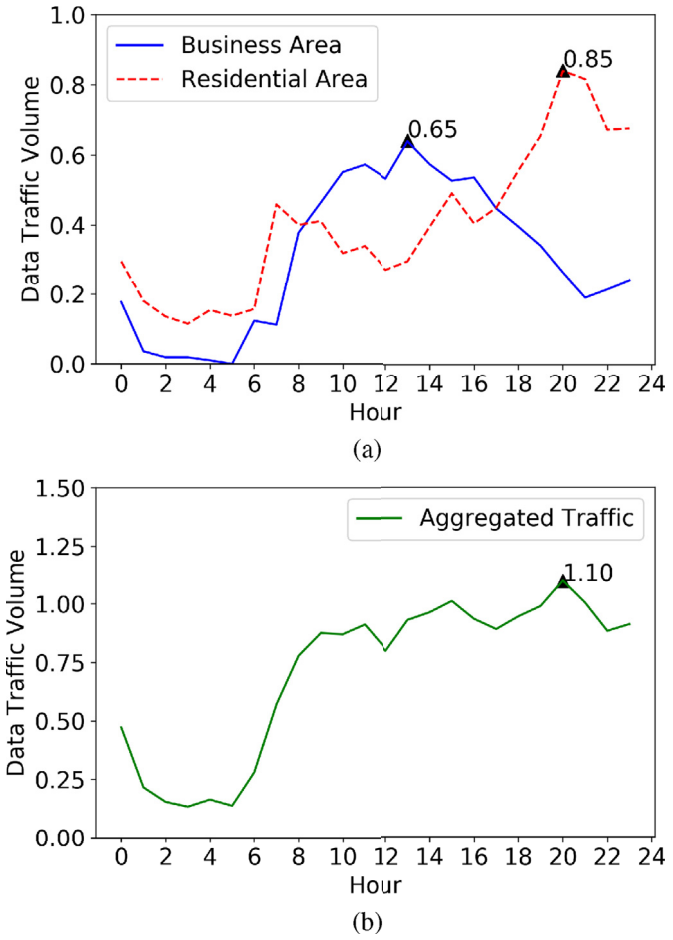


Fig. 1. (a) Data traffic patterns in different areas of Milan during a typical weekday. The blue solid line denotes the data traffic in a business district (Centro Direzionale), while the red dashed line corresponds to the data traffic in a residential area (Quintosolo District). (b) The aggregated data traffic pattern of the two areas. Triangles indicate the peak traffic hour and volume. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

non-uniform property of traffic demand poses great challenges for operators to optimize the capital expenditures and operating expenses of their network infrastructures. On one hand, the data processing capacity of each base station needs to cover its peak traffic volume, leading to high *deployment cost*. On the other hand, the capacity in individual base station is wasted during off-peak hours, resulting in low *capacity utility*.

Fortunately, with the rapid evolution of mobile network architectures, the emergence of *Cloud Radio Access Network (C-RAN)* (C. M. R. Institute, 2011) has presented new opportunities to address the above challenges. In C-RAN, a traditional base station is split into two components: a *Remote Radio Head (RRH)* for radio communication, and a *Baseband Unit (BBU)* for mobile data processing. The BBUs are further detached from the RRHs and hosted in centralized *BBU pools* (Checko et al., 2015). The RRHs and BBU pools are usually connected via high speed optical fiber (Checko et al., 2015). By clustering RRHs with *complementary traffic patterns* to a BBU, the data processing capacity in the BBU can be shared among RRHs in different time periods, and thus increasing the capacity utility of the BBU (Bhaumik et al., 2012). Furthermore, the required capacity of the BBU is expected to be smaller than the sum of capacities of single base stations, leading to a decrease in deployment cost. For example, in Fig. 1, if we cluster the RRHs in the business district (blue) and in the residential area (red) to a BBU, the aggregated traffic pattern will become relatively stable and the BBU will have a higher capacity utility (Fig. 1b). Meanwhile, the capacity required for the BBU can be reduced from the sum of the two peaks (1.50 = 0.65 + 0.85) to a lower aggregated value (1.10). In summary, by pooling BBUs from multiple base stations into a centralized BBU pool, the *statistical multiplexing gain* (Checko et al., 2015) can be achieved in the C-RAN architecture (C. M. R. Institute, 2011).

In order to unlock the power of the C-RAN architecture, it is of great importance to characterize the traffic patterns of RRHs, and to cluster complementary RRHs to a set of BBUs (Bhaumik et al., 2012; Chen et al., 2016a), so as to maximize the capacity utility and minimize the deployment cost. However, since the data traffic generated in the RRHs are *highly dynamic* over different time and locations, accurately foreseeing and characterizing the RRH traffic patterns in advance is quite challenging, hindering the optimization of RRH clustering and BBU mapping. More specifically, given a set of RRHs in a city, we need to accurately foresee their data traffic patterns in a future period of time (e.g., one day), and find optimal schemes to cluster RRHs with complementary traffic patterns, and map them to a set of BBUs for that period of time. In order to achieve these goals, we need to address the following issues:

1. **How to foresee the RRH traffic for a future period of time?** The data traffic in each RRH can vary significantly, depending on the impacts of temporal contexts (e.g., weekdays or weekends), human mobility, and social events, etc. Moreover, the data traffic of RRHs located in similar functional areas may demonstrate potential correlations. For example, during weekdays, the RRHs located in business districts usually observe data traffic peaks during working hours, and low data traffic volumes at nights. Capturing the *hidden temporal dependency and spatial correlation* among RRH traffic patterns is not trivial using state-of-the-art time series models, such as ARIMA (Hamilton, 1994) or neural networks (Zhang, 2003). Therefore, we need to foster more effective techniques for accurate RRH traffic pattern forecasting.
2. **How to measure the complementarity among RRHs?** In order effectively to share and reuse the capacity of a BBU mapped to a cluster of RRHs, the traffic peaks of the RRHs in the cluster should be scattered temporally (i.e., occur at different hours). Meanwhile, to make full use of the BBU mapped to a cluster and avoid BBU overloading, the aggregated cluster traffic should be close to the BBU capacity to a maximal extent, while not exceed the BBU capacity too much. Therefore, we need to take into account both aspects, i.e., *the peak distribution and the capacity utility*, to design an effective metric to measure the complementarity of RRHs.
3. **How to optimally cluster complementary RRHs into BBUs?** Given the traffic forecast and the complementarity measurements of RRHs, there are potentially enormous numbers of schemes to cluster these RRHs and map them to BBUs in a pool. The optimal scheme not only needs to maximize the average BBU capacity utility, but also needs to minimize the overall deployment cost. Moreover, in order to support fast handover and content offloading between neighboring RRHs (Checko et al., 2015; Zhao et al., 2016), the *distances* among a cluster of RRHs should be constrained within a reasonable range. Therefore, we need to design an effective algorithm to find the optimal RRH clustering scheme under the *distance constraint*.

With the above-mentioned research objectives and issues, the **main contributions** of this paper are:

- We propose a *deep-learning*-based approach to accurately foresee RRH traffic patterns for a future period of time. The proposed approach is capable of modeling the temporal dependency and spatial correlation among the RRH data traffic, and accurately forecasting the future traffic pattern based on the historical observations.
- We propose a two-phase framework to dynamically find optimal

RRH clustering and BBU mapping schemes under different contexts. In the first phase, we forecast the traffic patterns of RRHs leveraging the proposed MuLSTM model, and propose an *entropy-based metric* to characterize the *complementarity* of RRHs, taking into account both the peak distribution and capacity utility. In the second phase, we build a *weighted graph* to model the complementarity of RRHs, and propose a *distance-constrained clustering algorithm* to find optimal RRH clustering schemes with the objectives of both capacity utility and deployment cost.

- We evaluate the performance of our method using datasets in two months from real-world mobile networks in Milan and Trentino, Italy. Results show that our method effectively increases the average capacity utility to 83.4% and 76.7%, and reduces the overall deployment cost to 48.4% and 51.7% of the traditional RAN architecture in the two datasets, respectively, which consistently outperforms the state-of-the-art baseline methods.

The rest of this paper is organized as follows. We first present a literal review in Section 2, and then introduce the preliminaries and the proposed framework in Section 3. In Section 4 we propose the deep-learning-based dynamic RRH profiling method, and in Section 5 we propose the graph-based complementary RRH clustering algorithm. We report the evaluation results and present case studies with real-world datasets in Section 6. Finally, we conclude our work in Section 7.

2. Related work

2.1. Cloud Radio Access Network

Cloud Radio Access Network (C-RAN) is a novel mobile network architecture to address the challenges faced by operator while trying to meet the fast-growing traffic demand. The details of the C-RAN concept can be found (C. M. R. Institute, 2011). The basic idea of C-RAN is to pool the data processing units from multiple RRHs into centralized BBU pools, so that the pool capacity can be shared among these RRHs. Since fewer BBUs are needed and higher BBU capacity utility can be achieved, the C-RAN architecture can reduce the network deployment cost and energy consumption (Checko et al., 2015). Therefore, C-RAN is seen as a typical architecture of the fifth generation (5G) network in the year 2020 horizon (I et al., 2014).

One of the key problem in the C-RAN architecture is to a design optimal RRH clustering scheme and connect them to the BBU pool. An optimal scheme should facilitate the BBU capacity utility in the pool, reduce the deployment cost, and also prevent the propagation delay between RRHs and BBU pool (Checko et al., 2015). To this end, Bhaumik et al. (2012) proposed CloudIQ, a framework for partitioning a set of RRHs into groups and process the signals in a shared data center. Since the distance between data centers and the RRHs may lead to potential delay between distant RRHs and the data center (Checko et al., 2015). Lee et al. (2013) proposed a RRH cooperation scheme with dynamic clustering in C-RAN, however the objective of the cooperation is to derive the signal-to-interference for RRH evaluation. One of the very relevant ideas to our work was illustrated in (Zheng et al., 2016), which explored approaches to integrate big data analytics with network optimization in 5G, especially by exploiting historical data to optimize resource allocation in centralized BBUs in C-RAN.

2.2. Time series forecasting models

During the past decades, time series modeling and forecasting have been extensively studied in the literature (Hamilton, 1994; Dorffner, 1996; Zhang, 2003). In this section, we survey two of the state-of-the-art approaches in time series analytics, and discuss their disadvantages in addressing our problem.

Autoregressive Integrated Moving Average (ARIMA) models: In time series analysis, ARIMA models are commonly used to fit a time series data and to forecast future variations in the series. ARIMA models explicitly extract from a time series three intuitive features, i.e., *auto-regression, moving average, and integration*. The auto-regression (AR) part indicates that the evolving variable of a time series is regressed on its own lagged values. The moving average (MA) part indicates that the regression error can be represented as a linear combination of error terms dependent on the values in the past. The integration (I) part is applied to the regression model to represent non-stationary time series (i.e., the variable in the time series shows a trend of increasing or decreasing). ARIMA models are capable of rapidly adjusting for sudden changes in trend, and it has been proved successful in many short-term forecasting problems (Sang and Li, 2002). However, for long-term forecasting problems which involve predicting multiple future steps, the error of ARIMA models *accumulate* significantly and the forecasting confidence *decrease* rapidly as the forecasting step grows (Box et al., 2015). In our problem, we need to accurately forecast the RRH traffic for several hours to foresee the traffic patterns in the future for RRH clustering, which poses great challenges for the ARIMA models.

Artificial Neural Network (ANN) models: Recently, ANN models are widely employed to understand time series and forecast the future trend by leveraging a sliding-window-based technique (Dorffner, 1996), which can be named *windowed-ANN*, or WANN. More specifically, this technique first slices a time series into several equal-length windows, and then feeds these windows into an ANN model as *features*. The *output* of the model is the forecast of the future values of the time series, which can either be short-term or long-term results, depending on the application scenario. The WANN models have been applied in various domains, such as financial market (Azoff, 1994) and operation research (Zhang and Qi, 2005). However, one of the biggest problem of the WANN model is its incapability to model the *temporal dependency* between the elements in each time series window. In fact, the elements in a window is treated equally as input features and thus the *sequential order* of the elements is ignored. As a result, the WANN model can make fluctuating and inconsistent forecasts which are not desired in our problem.

In this work, we propose a deep-learning (LeCun et al., 2015) architecture to model the temporal dependency of RRH traffic and the spatial correlations among RRHs in a unified framework. Such kind of spatial-temporal deep-learning framework has been widely used in IP and transportation network traffic prediction (Nie et al., 2016; Zhang et al., 2016), electronic health records understanding (Rajkomar et al., 1801), and social network behavior analytics (Zhang et al., 2017).

2.3. Mobile data analytics

With the emergence of ubiquitous sensing and computing diagrams (Zhang et al., 2011), a massive number of mobile data can now be collected either by mobile crowdsensing paradigms (Wang et al., 2016, 2017; Guo et al., 2015) or from operators' infrastructures. These heterogeneous mobile big data are being extensively analyzed in the literature to retrieve interesting and informative information (Chen et al., 2014, 2016b; Yang et al., 2015; Tan et al., 2016). For example, Barlacchi et al. (2015) released a large-scale Call Detail Records (CDR) dataset from Telecom Italia, containing two-months of calls, SMSs and network traffic data from the city of Milan and Trentino, Italy. Based on the dataset, Furno et al. (2016) proposed a data analytics framework to builds profiles of the city-wide traffic demand, and identifies unusual situations in network usages, aiming at facilitating the design and implementation of cellular cognitive networking. Cici et al. (2015) studied the decomposition of cell phone activity series, and connect the decomposed series to socio-economic activities, such as regular working patterns and opportunistic events (Chen et al., 2017b).

However, applying real-world mobile network data to C-RAN optimization has not yet been extensively studied in the literature, since previous works mainly focus on simulation-based approaches to model network traffic (Zhan and Niyato, 2017; Zhang et al., 2016). In this

work, we exploit large-scale open datasets from real-world mobile network operators to understand the traffic patterns in real networks, and then conduct C-RAN optimization studies based on the knowledge discovered from these mobile datasets.

3. Preliminaries and framework

3.1. Preliminaries

In mobile network architectures, a set of base stations are deployed over geographical areas called cells (Tse and Viswanath, 2005). Each base station provides the cell with the network coverage which can be used for transmission of voice and data. With the recent emergence of smartphones and tablets, the data traffic generated from users connected to the RRHs is increasing rapidly (Cisco, 2016; J. Research, 2011).

In order to benchmark the data processing capacity of base stations, many operators have collected large scales of RRH traffic statistics data and make them publicly available (Zheng et al., 2016). In this paper, we exploit the dataset released by Telecom Italia for the Big Data Challenge initiative (Barlacchi et al., 2015). We extract two months of network traffic data from 11/01/2013 to 12/31/2013 in the city of Milan, Italy and the province of Trentino, Italy. We also collect the locations of active base stations in Milan and Trentino during the two months from CellMapper.net,[1] and derive the traffic volume of each base station during the two months on an hourly basis. The traffic data pre-processing steps will be detailed in the evaluation section.

In this work, we consider a C-RAN architecture with one BBU pool for the city-wide mobile network. The benefits of adopting such a centralized pool are two-fold. First, the deployment cost and energy consumption can be greatly reduced by employing data center virtualization technologies (Qian et al., 2015). Second, the handover handing and contents offloading among RRHs can be processed internally in the pool, which significantly reduces delays and increases throughput (Checko et al., 2015). BBUs in the pool are implemented as virtual machines with specific predefined capacities. In this work, for fair of comparison and simplicity, we assume the BBU capacity to be *fixed and equal to* the on-site BBUs in the traditional architecture. We discuss the implement details in the evaluation section.

3.2. Framework overview

We propose a two-phase framework to dynamically cluster complementary RRHs to a set of BBUs, so that the BBU capacity utility and the deployment cost of the entire network can be optimized. As shown in Fig. 2, in the dynamic RRH profiling phase, given a set of RRHs at a time point, we first propose a deep-learning-based approach to forecast the traffic patterns of RRHs in a future period of time based on their historical traffic data, and then calculate the complementarity of RRHs using a proposed entropy-based metric. In the dynamic RRH clustering phase, we first build a graph model to represent the complementarity among RRHs, and then propose a distance-constrained clustering algorithm to cluster RRHs with complementary traffic patterns. We elaborate on the details of this framework in the following sections.

4. Dynamic RRH profiling

In order to cluster RRHs with complementary traffic patterns to a BBU, we need to be able to forecast the traffic pattern of each RRH for a future period of time. Since the traffic of RRHs vary significantly and exhibit spatial correlations, we propose a deep-learning-based approach to model the spatial-temporal dynamics and to forecast the future traffic pattern accurately. Based on the traffic forecast, we dynamically characterize the complementarity of RRHs, focusing on the peak distribution and capacity utility of a cluster of RRHs, and design an entropy-based metric to characterize their complementarity.

4.1. RRH traffic forecasting

Based on the historical traffic data, we observe that the traffic patterns of RRHs are highly dynamic under different temporal contexts. For example, Fig. 3 shows the traffic patterns of two RRHs located in two business districts in Milan during one week, respectively. We observe significant traffic peaks during the working hours of weekdays, and low capacity utility during off-work hours. Moreover, we observe that the traffic patterns of RRHs located in similar functional areas usually demonstrate similar trends. For example, in Fig. 3, the traffic patterns in the two business districts of Milan show similar weekday-weekend patterns.

4.1.1. Basic idea

In order to accurately forecast the traffic patterns of the RRHs in a future period of time, we need to be able to effectively capture their temporal dependency and spatial correlation. However, this is not trivial using the state-of-the-art techniques. In this work, we propose a *deep-learning*-based approach for our problem. More specifically, we exploit the Recurrent Neural Network (RNN) to automatically capture the intrinsic temporal dependency in our traffic data. An RNN is a special type of neural network designed for sequential pattern mining problems (Sutskever et al., 2014). Built upon the windowed-ANN architecture, an RNN features additional loops to the neurons in the layers of the neural network. Each neuron may pass its signal laterally in addition to forward to the next layer, and consequently, the output of the network for a window may feedback as an input to the network for the next window. Such *recurrent connections* add state or memory to the windowed-ANN architecture and allow it to learn and harness the intrinsic temporal dependency in the time series.

Unfortunately, training an RNN effectively is technically challenging due to the *vanishing or exploding gradient problem*, i.e., the weights in the training procedure quickly became so small as to have no effect (vanishing gradients) or so large as to result in very large changes (exploding gradients). To overcome this problem, researchers proposed the Long Short-Term Memory Network (LSTM) model (Gers et al., 2002), which introduces the concepts of memory cells and forget gates to generate consistent data flow between the layers of the network and keep the weights stable (Hochreiter and Schmidhuber, 1997). In this work, we exploit the LSTM model to effectively learn the temporal dependency of our traffic data.

The other challenge is to model the *spatial correlation* between RRHs in the network. The above-mentioned approaches typically model the traffic of each RRH as a separate time series, making it difficult to capture the correlation between RRHs. In this work, we propose a multivariate-Long Short-Term Memory Network (MuLSTM) approach to model the RRH traffic in a city in a unified model, putting each RRH traffic as a sequence for training and forecasting, and consequently learn the spatial correlation between RRHs.

4.1.2. The MuLSTM model

Before introducing the MuLSTM model, we define several important terminologies as follows:

Definition 1. Remote Radio Head *(RRH)*:

The RRHs in a city-wide mobile network can be described as a set of points denoted by the following 3-tuple:

$$\{r|r = (rid, lat, lng)\}$$

where rid, lat, lng are the unique ID, latitude, and longitude of the RRH.

Definition 2. *RRH Traffic*:

The mobile data traffic collected from each RRH can be denoted by a set

[1] https://www.cellmapper.net/map.

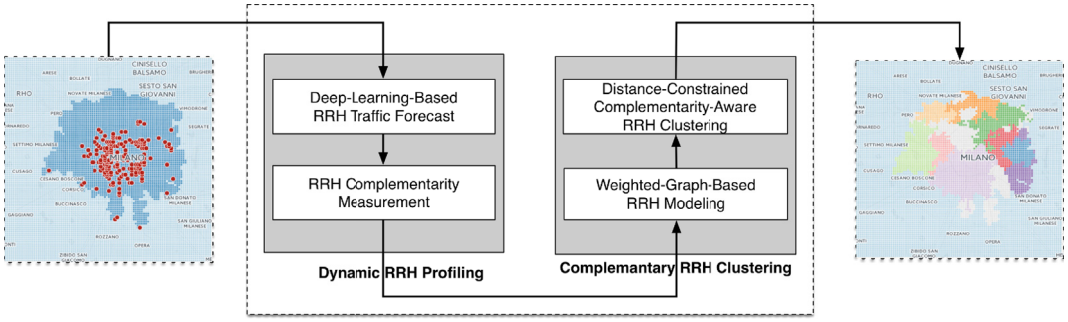
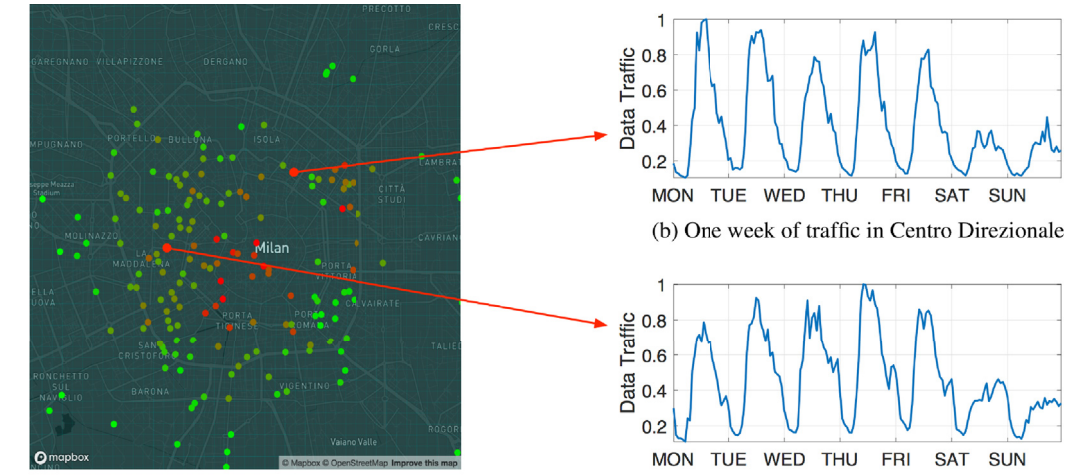


Fig. 2. Framework overview.



(a) Base stations in Milan (11/01/2013 - 12/31/2013)

(b) One week of traffic in Centro Direzionale

(c) One week of traffic in Piazza Wagner

Fig. 3. The locations of base stations in Milan and two of the illustrative examples of traffic patterns observed in two business districts from 11/25/2013 to 12/01/2013. Red color denotes high average traffic volume and green color corresponds to low average traffic volume. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

of fixed-length sequences:

$$\{f|f_i = [u_i(1), \ldots, u_i(t), \ldots, u_i(N_t)]\}$$

where $u_i(t)$ is the traffic volume of RRH i in time span t($1 \le t \le N_t$). In this work, we use a one hour time span.

With the collected traffic data, we first organize the collected RRH traffic into a matrix $F^{R_{N_t} \times R_{N_r}}$, where N_t is the number of time spans, and N_r denotes the number of RRHs in the network. We denote the traffic of RRHs we have observed until time t as $F([0, t],:)$, and the traffic of RRHs we would like to forecast in a future period of time Δt as $F([t, t + \Delta t],:)$. In this work, to simplify the implementation, we use one hour time span, and $\Delta t = 24$ h with $t \bmod 24 = 0$, i.e., we forecast the hourly traffic of RRHs for the next day at the end of each day, and dynamically update the RRH clustering scheme based on the forecast. Based upon this, we generate a set of *traffic snapshots* from the traffic matrix, which is defined as follows.

Definition 3. ***RRH Traffic Snapshot***:

A traffic snapshot is defined as a matrix F_i, which corresponds to the traffic of all the RRHs during a given period of time Δt, i.e.,

$$\mathscr{F} = \{F_i|F_i = F([(i-1)*\Delta t, i*\Delta t], :),\ i = 1, 2, \ldots\}$$

In order to make traffic forecast, we train a *sequence to sequence model* (Sutskever et al., 2014) leveraging a unified multivariate LSTM model. During each forecasting, the model accepts F_i as input and outputs F_{i+1}. Note that such a model is called a *many-to-many sequential* model because both the input and output contain Δt time spans, and the order of the time spans play an important role in shaping the model's inner structure. Moreover, the traffic of RRHs are input to the model as *multivariate features* simultaneously, which enables the model to learn the spatial correlation between RRHs.

Finally, we elaborate on the design of the MuLSTM network structure. In general, the MuLSTM model follows the encoder-decoder structure by stacking two LSTM layers L_1 and L_2. The encoder L_1 accepts a snapshot of size $[\Delta t, N_r]$, learns the temporal and spatial structures in the snapshot, and passes the encoded sequences to the decoder. The decoder then makes forecast for a future snapshot of size $[\Delta t, N_r]$ based on the learned structures. The model is trained using the popular Backpropagation Through Time (BPTT) algorithm for multiple iterations. We elaborate the details of the model parameters in the evaluation section.

4.2. RRH complementarity measurement

Once we have the traffic snapshot forecast for the next day, we are able to evaluate the complementarity of RRHs in that context, and cluster complementary RRHs to a BBU. We consider the following two aspects to design an effective complementarity metric of RRHs.

4.2.1. Peak distribution

The peak traffic volume of a set of RRHs clustered to the same BBU should be scattered in different temporal contexts, so that the capacity of the BBU can be shared among these RRHs. To this end, we design an entropy-based metric to measure the peak distribution of a set of RRH. Specifically, given a set of clustered RRHs $C = \{r_1, \ldots, r_n\}$, we first find the peak hours in their traffic profiles, respectively, i.e.,

$$T(r_i) = \{t_{i_1}, t_{i_2}, ..., t_{i_m}\}, \quad 1 \le i_m \le 24 \tag{1}$$

where t_{i_m} denotes the m_{th} peak time of r_i. Then, we calculate the *Shannon entropy* (Lin, 1991) of the peak hours of the set of clustered RRHs $T(C) = \cup T(r_i)$ as follows:

$$H(C) = -\sum_{k=1}^{K} p_k \log p_k \tag{2}$$

where $K = |T(C)|$ corresponds to the total quantity of peaks in C, and p_k is the probability of observing the corresponding peak hour in the set $T(C)$. A larger entropy value of a RRH cluster indicates that the RRHs are more complementary in the cluster w.r.t. traffic patterns.

4.2.2. Capacity utility

To make full use of the BBU mapped to a cluster C, the aggregated cluster traffic should be close to the BBU capacity in different hours of the day. Meanwhile, to prevent the BBU from overload, the aggregated cluster traffic should not exceed the BBU capacity too much. To this end, we design the following metric to quantitatively measure the capacity utility of a BBU B mapped to a cluster C:

$$U(C) = \left(\frac{\text{mean}\mathbf{f}(C)}{|B|}\right)^{-\ln \frac{\text{mean}\mathbf{f}(C)}{|B|}} \tag{3}$$

where $\mathbf{f}(C) = \sum_{i=1}^{n} \mathbf{f}(r_i)$ denotes the aggregated traffic profile of the RRH cluster, and $|B|$ is the fixed BBU capacity measured in traffic volume. Fig. 4 shows the curve of the capacity utility function, which achieves its maximal when the mean aggregated traffic volume is equal to the BBU capacity.

Finally, we calculate the complementarity of the RRH cluster C as follows:

$$M(C) = U(C) * H(C) \tag{4}$$

$$= -\left(\frac{\text{mean}\,\mathbf{f}(C)}{|B|}\right)^{-\ln \frac{\text{mean}\,\mathbf{f}(C)}{|B|}} \sum_{k=1}^{K} p_k \log p_k \tag{5}$$

5. Complementary RRH clustering

In this phase, our objective is to cluster RRHs with complementary traffic patterns to a set of BBUs in a pool. One intuitive method is to exhaustively search for RRHs with complementary traffic patterns and iteratively cluster them. However, since there are a tremendous number of clustering schemes, such a method can be computationally intractable as the network scale increases. Moreover, the distance between RRHs and BBU pool should also be constrained within a range, since the propagation delay between RRHs and BBU pool may exceed quality-of-service requirements as distance increases, and we also need to enable machine to machine communications between RRHs such as handover (Tekinay and Jabbari, 1991) in the mobile network.

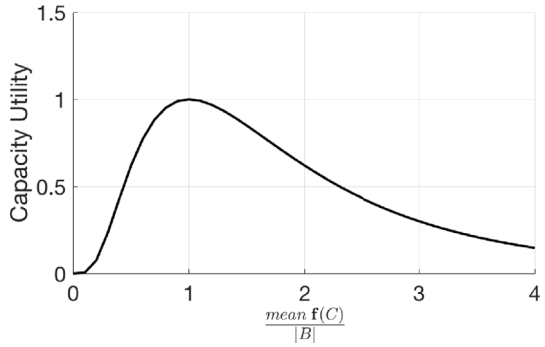


Fig. 4. The curve of the designed capacity utility function, which reaches its maximal when the cluster traffic volume equals the BBU capacity.

Table 1
Dataset description.

Item	Milan	Trentino
# Grids	10,000	11,466
Grid size	55,225 m^2	1000,000 m^2
# RRH	182	522
# Covered grids	2918	2035
Average coverage	885,420 m^2	3,932,950 m^2
Average traffic volume	0.19	0.13
Data collection period	11/01/2013–12/31/2013	

Therefore, we propose a graph-model-based algorithm to effectively cluster neighboring RRHs to the same BBU under distance-constraints. First, we construct a weighted graph model to represent the relationship of RRHs, exploiting graph links to express the RRH distance constraints, and link weights to characterize the RRH complementarity measurement. Then, we propose a community-detection-based algorithm to iteratively cluster RRHs into clusters, so that the complementarity of RRHs is maximized within each cluster and minimized across different clusters.

5.1. Weighted-graph-based RRH modeling

We model the complementarity among RRHs as an undirected, weighted graph $G = (V, E)$, where $V = \{r_1, \ldots, r_N\}$ denotes the set of N RRHs, and E denotes the set of links between two RRHs.

We then define the adjacency matrix A of graph G, which is an $N \times N$ symmetric matrix with entries $a_{i,j} = 1$ when there is a link between RRH r_i and RRH r_j, and $a_{i,j} = 0$ otherwise ($i, j = 1, \ldots, N$). We use the geographic distance of two RRHs to determine whether they are adjacent or not. More specifically, for RRH r_i and RRH r_j, we define:

$$a_{i,j} = \begin{cases} 1, & \text{if } dist(r_i, r_j) \le \tau \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $dist(r_i, r_j)$ is the geographic distance between the two RRHs, and τ is a *neighborhood threshold* controlling the geographic distance of neighboring RRHs.

Given two neighboring RRHs, we use their complementarity measurement to determine their link weight, i.e.,

$$w(r_i, r_j) = M(\{r_i, r_j\}) * a_{i,j} \tag{7}$$

We consider the case of normalized symmetric positive weights ($w(r_i, r_j) \in [0, 1]$) with no loops ($w(r_i, r_i) = 0$). We note that $w(r_i, r_j) = 0$ when there is no link between r_i and r_j ($a_{i,j} = 0$).

5.2. Distance-constrained RRH clustering

In this step, we need to cluster RRHs to a BBU, so that each cluster consists of neighboring RRHs with complementary traffic patterns. As the link weight of graph G encodes the complementary of RRHs, we need to cluster RRHs with high link weights together, which can be identified as a community detection problem (Newman and Girvan, 2004).

Problem: Given graph $G = (V, E)$, we first define a set of clusters $\mathbb{P} = \{C_1, ..., C_K\}$, where

$$\cup_{\forall C_k \in \mathbb{P}} = V \quad \text{and} \quad \cap_{\forall C_k \in \mathbb{P}} = \emptyset \tag{8}$$

Then, given a RRH v, we define the *connectivity* of v to a cluster C as the sum of link weights between v and the RRHs in the cluster C:

$$con(v, C) = \sum_{v' \in C} w_{v,v'} \tag{9}$$

Finally, we define the *adjacent clusters* $\mathbb{C}(v)$ of v as

$$\mathbb{C}(v) = \{C | con(v, C) > 0, C \in \mathbb{P}\} \tag{10}$$

Table 2
Evaluation results.

Methods	Traffic Forecast Error (MAE)		Average Capacity Utility		Overall Deployment Cost	
	Milan	Trentino	Milan	Trentino	Milan	Trentino
Traditional	–	–	38.8%	29.4%	182	522
ARIMA-DCCA	0.202	0.237	65.3%	45.2%	112	160
WANN-DCCA	0.175	0.198	73.4%	58.8%	96	120
MuLSTM-DC	0.074	0.083	58.7%	39.2%	120	180
MuLSTM-DCCA (Proposed)	**0.074**	**0.083**	**83.4%**	**76.7%**	**88**	**270**

With the above definition, our objective is to find an optimal set of clusters $\mathbb{P}$, so that the internal connectivity within a cluster is higher than the inter-cluster connectivity, i.e.,

$$\forall\ v \in C_k,\ \ con(v, C_k) \geq \max\{con(v, C_l), C_l \in \mathbb{P}\} \tag{11}$$

We also need to bound the distance span of a cluster within the neighborhood threshold, i.e.,

$$\forall\ v, v' \in C_k,\ \ dist(v, v') \leq \tau \tag{12}$$

Solution: Based on the label propagation concept (Chen et al., 2016a; Raghavan et al., 2007), we propose a *Distance-Constrained Complementarity-Aware (DCCA)* algorithm to cluster RRHs. The basic idea of DCCA is iteratively assigning RRHs to the adjacent clusters, where the *gain* of assigning RRH v to cluster C is iteratively evaluated by a *value function* as follows:

$$value(v, C) = con(v, C) \times \log\left(\frac{\tau}{\max\{dist(v, v')\}}\right) \tag{13}$$

The DCCA algorithm greedily assigns the RRHs to the adjacent cluster with highest value[2] until none of the RRHs are moved among clusters (Raghavan et al., 2007). As the convergence of such a greedy approach is difficult to prove, we set a maximum iteration number max _*iter* to ensure the algorithm will stop.

Algorithm: The DCCA algorithm is initialized by assigning each RRH in the graph to a unique cluster label. In each iteration, we randomly populate a list of RRH $\mathscr{L}$, and traverse the list to update the cluster label of each RRH. The label update process is as follows. First, we remove the RRH from its current cluster, and find the set of adjacent clusters to the current RRH. Then, we compute the value function for all the adjacent clusters, and assign the RRH to the cluster with the highest value. We mark the RRH as *moved* among clusters if its new cluster label is different from the old one. After we finish iterating over the RRH list, we decide whether to perform another iteration or finish the algorithm based on the following stop criteria: (1) the specified maximum iteration number max_*iter* is reached, or (2) none of the RRH are moved among clusters.

6. Evaluation

In this section, based on a real-world mobile network traffic dataset, we evaluate the performance of our framework by assessing its ability to reduce deployment cost and energy consumption. We first describe the experiment settings, and then present the evaluation results and case studies.

6.1. Experiment settings

Datasets: The Telecom Italia Big Data Challenge dataset (Barlacchi et al., 2015) contains two months of network traffic data from 11/01/2013 to 12/31/2013 in Milan and Trentino, Italy, respectively. The city of Milan is partitioned into 100×100 grids with grid size of about 235×235 square meters, while the province of Trentino is partitioned into 117×98 grids with grid size of about $1,000 \times 1,000$ square meters. In each grid, the traffic volume is recorded on an hourly basis. We compile a base station dataset from CellMapper.net, which consists of the locations and coverage areas of active base stations observed in the two months. Based on the location and coverage of each base station, we find the corresponding covered grids and calculate their traffic volume. Finally, we normalize the traffic volumes of each base station to the [0, 1] range for the convenience of analytics. The details of these two datasets are listed in Table 1.

BBU Capacity: We determine the BBU capacity based on the normalized traffic volume. For the traditional architecture, we assume that each RRH is equipped with an on-site BBU with a capacity of one normalized traffic volume. In this way, the traffic in each RRH can be covered by the BBU. We define the capacity of the on-site as a *capacity unit*. For the C-RAN architecture, we assume that the BBUs in the pool (pool BBU) are of the same size, and the capacity is of Q ($Q = 1, 2, \ldots$) capacity unit, so that the traffic of a cluster of RRHs traffic can be handled in a BBU without causing significant overload. In this work, based on a series of empirical experiments, we choose $Q = 8$ for the city of Milan, and $Q = 10$ for the province of *Trentino*, respectively.

Evaluation Plan: Based on the collected datasets, we map the grids to the coverage areas of RRHs, and aggregate the traffic data to the corresponding RRHs on an hourly basis. We then generate a set of 61 daily traffic snapshots $\mathscr{F}$, each containing the 24 h' traffic for all the 182 RRHs. We use the snapshots of the first 70% as the training set $\mathscr{F}_{train}$, and the snapshots of the remaining 30% as the test set $\mathscr{F}_{test}$. For the test set, we calculate the complementarity of RRHs based on the traffic forecast, and construct a graph of 182 nodes with the corresponding link structure based on the complementarity metrics. Finally, we perform the DCCA algorithm to cluster the complementary RRHs to a set of BBUs in a centralized pool.

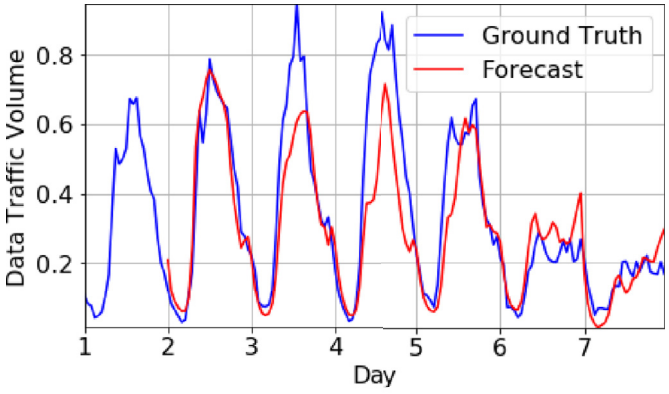
Model Specification: We construct a MuLSTM model with two stacked LSTM layers. The encoder layer L_1 contains $N_{encoder}$ memory units, which accepts a traffic snapshot of shape [24, 182] as input, and outputs an encoded sequence for the decoder. The decoder contains $N_{decoder}$ memory units, which accepts the encoded sequence as input and outputs the forecast of the traffic snapshot. We train the network with the training set $\mathscr{F}_{train}$ for N_{iter} iterations to ensure that the network learns the potential temporal and spatial structures.

Model Training: We use the popular Tensorflow (Abadi et al.,) library for constructing our deep-learning model. Based on a series of empirical experiments, we choose the optimal $N_{encoder} = N_{decoder} = 32$, and $N_{iter} = 10,000$. The model is trained on a 64-bit server with an NIVIDA GeForce GTX 1080 graphic card and 16 GB of RAM. Each training iteration takes about 3 s and the whole process takes 8.3 h.

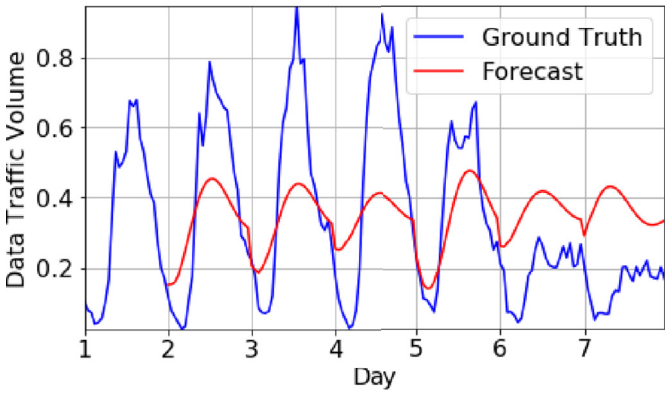
Evaluation Metrics: We design the following evaluation metrics to evaluate the RRH traffic forecasting phase and the RRH clustering phase respectively.

(1) For the RRH traffic forecasting phase, we compare the traffic snapshot forecast $\hat{F}_i$ with the ground truth data F_i in the test set, and calculate the Mean Absolute Error (MAE) for each snapshot:

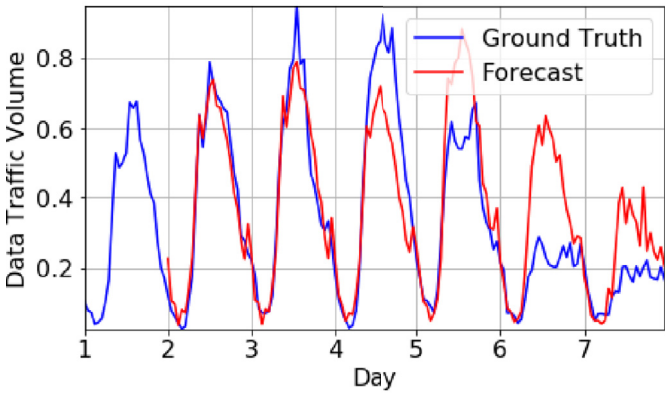
[2] If two clusters yield the same value, we randomly choose one.



(a) MuLSTM forecast



(b) ARIMA forecast



(c) WANN forecast

Fig. 5. RRH traffic forecast results for the base station located in a business district (Centro Direzionale) from 12/25/2013 to 12/01/2013 (one week). The first day traffic is used for input and thus there is not prediction.

$$MAE(F_i, \hat{F}_i) = \frac{\sum_{t=1}^{N_t} \sum_{r=1}^{N_r} |F_i(t, r) - \hat{F}_i(t, r)|}{N_r \times N_t}$$

(2) For the RRH clustering phase, we quantitatively measure the statistical multiplexing gain from two aspects, i.e., the increase of *average capacity utility* and the decrease of *overall deployment cost*, compared with the on-site BBUs in traditional architecture. In order to measure the capacity utility of a clustering scheme $\mathbb{P} = \{C_1, ..., C_K\}$, we derive the following metric based on Equation (3), i.e.,

$$Utility(\mathbb{P}) = \text{mean}_{C_k} U(C_k) \tag{14}$$

based upon this, we calculate the average capacity utility of the test set. In order to measure the overall deployment cost, we sum up the total BBU capacity units required in the pool for a clustering scheme $\mathbb{P}$, i.e.,

$$Cost(\mathbb{P}) = \sum_{k=1}^{K} \left| \{C_k\} \right| \tag{15}$$

We use the maximal quantity of capacity units measured in the test set as the overall deployment cost required in the pool.

Baseline Methods: We design the following baseline methods to compare with the proposed method.

- **Traditional**: In the traditional architecture, one RRH is equipped with one on-site BBU with one capacity unit. The traffic forecast and RRH clustering is not necessary and thus not performed.
- **ARIMA-DCCA**: This baseline method uses the traditional ARIMA model for RRH traffic forecasting, one RRH at a time, and then use the proposed GCLP algorithm for RRH clustering.
- **WANN-DCCA**: This baseline method uses a windowed-ANN model for RRH traffic forecasting, which inputs a traffic snapshot for a day and outputs a traffic snapshot for the next day. The RRH clustering algorithm is the same as the proposed method.
- **MuLSTM-DC**: This baseline method uses the proposed MuLSTM model for RRH traffic forecasting, and then employs a *distance-constrained (DC)* clustering algorithm that clusters neighboring RRHs without considering their traffic complementarity. The clustering steps are similar to the propose DCCA method.

6.2. Evaluation results

Overall Results: Table 2 shows the overall evaluation results of the proposed method as well as the baseline methods. For the RRH traffic forecast accuracy, we can see that the proposed Mu-LSTM model achieves the lowest mean absolute error score (0.074 in Milan and 0.083 in Trentino) compared with the two baselines (ARIMA and WANN), validating its capability of modeling the temporal dependency and spatial correlation of RRH traffic and make accurate forecast. In contrast, the ARIMA method does not capture the spatial correlation among RRHs, while the WANN method is not capable of modeling the temporal dependency of RRH traffic patterns. Consequently, the two baselines have higher forecast error rate in both datasets.

For the RRH clustering results, the proposed method consistently achieves the highest average capacity utility (83.4% in Milan and 76.7% in Trentino), as well as the lowest overall deployment cost (88 capacity units in Milan and 270 capacity units in Trentino). Compared with the traditional architecture with on-site BBUs, the clustering schemes increase the average capacity utility rate from 38.8% to 83.4%, and reduce the overall deployment cost from 182 capacity units to 88 capacity units (48.4% of the original cost) in Milan, validating the possibility of achieving significant statistical multiplexing gain though C-RAN optimization. In comparison, the distance-constrained (MuLSTM-DC) clustering baseline does not consider RRH traffic complementarity in the optimization process, and thus are not able to increase capacity utility and decrease deployment cost as effective as the proposed method. Due to inaccurate traffic forecast results, the ARIMA-DCCA and WANN-DCCA baseline methods tend to produce suboptimal clustering schemes and thus achieving lower statistical multiple gain.

We also note that our method performs better in the city of Milan than in the province of Trentino, which can be explained by the geographic characteristic of Trentino. Specifically, Trentino is a mountainous region where cities and villages scatter among valleys. The RRHs are scattered distantly, making it difficult to form complementary RRH clusters in their neighborhoods. In contrast, the metropolitan areas of

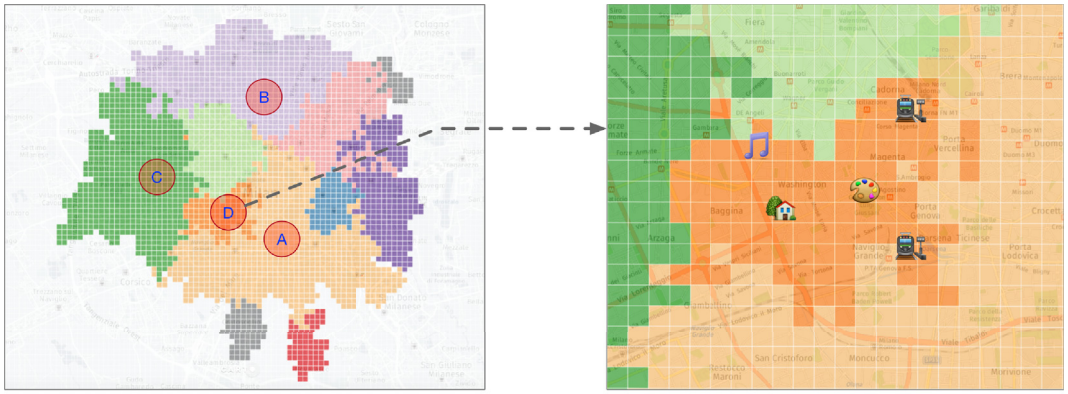


Fig. 6. (a) An illustrative example of RRH clustering scheme on 2013/11/25 (Monday) in Milan using the proposed method. Each colored area denotes a cluster and its corresponding coverage area. (b) Cluster D in details: a hybrid area with diverse traffic patterns. Icons denote the featuring city functions in this area.

Milan are larger, more concentrated and more populated, making it easier to form complementary clusters for C-RAN optimization.

Case Studies: We conduct some case studies in Milan to showcase the effectiveness of our method. For RRH traffic forecasting, Fig. 5 shows an illustrative example of the forecasting results using the proposed MuLSTM method as well as the ARIMA and WANN baseline methods. We can see that our method accurately forecasts the weekday and weekend traffic patterns based on the temporal dependency and spatial correlation it learns from the training set. Instead, the ARIMA method fails to learn the hybrid temporal dependency patterns and outputs the averaged traffic forecast. The WANN method is able to learn some hidden temporal dependency from the single RRH data but is not stable (e.g., on Friday and Saturday).

Fig. 6 shows the RRH clustering scheme with the proposed method on 2013/11/25 (Monday) in Milan. In general, we obtain 12 RRH clusters, each connected to a BBU in the centralized pool. In Fig. 6a, we can see that many clusters (e.g., Cluster *A*, *B*, and *C*) are composed of an urban part and a suburban part, indicating that the traffic patterns in these areas are potentially complementary during a typical weekday. We also note that cluster *D* is concentrated in a relatively small area, indicating the diverse traffic patterns within this area (Fig. 6b). The reason is probably due to the hybrid functions of this area, which consists of a large residential district (the *Washington neighborhood*), several national museums and theaters (e.g., *Museo Nazionale Scienza e Tecnologia Leonardo da Vinci* and *Teatro Nazionale CheBanca*), and a transportation hub consisting of several train and metro stations (*e.g., Milano Porta Genova* and *Milano Cadorna*). The algorithm is able to identify the RRHs with complementary traffic patterns during the day and effectively cluster them into a BBU to achieve statistical multiplexing gain.

7. Conclusion

In this work, we focus two of the most important objectives in C-RAN optimization to achieve statistical multiplexing gain, i.e., increasing capacity utility and reducing deployment cost. Accordingly, we proposed a deep-learning-based framework to achieve these goals in C-RAN optimization. Specifically, we forecast the traffic patterns of RRHs using a multivariate LSTM model, and then cluster complementary base stations to BBUs based on the traffic patterns. The proposed MuLSTM model is capable of modeling the temporal dependency and spatial correlation between RRHs in the network, and the proposed DCCA clustering algorithm is effective in finding optimal clustering schemes under certain distance constraints, with the objectives of both maximizing the capacity utility and minimizing the deployment cost. Real-world evaluation results in Milan and Trentino show that our framework effectively increases the average capacity utility to 83.4% and 76.7%, and reduces the overall deployment cost to 48.4% and 51.7% of the traditional RAN architecture in the two datasets, respectively, which consistently outperforms the state-of-the-art baseline methods.

In the future, we plan to improve this work in the following directions. Firstly, we plan to explore the variations in the BBU pool, such as considering different sizes of BBU capacity. Secondly, we plan to evaluate our framework in more datasets, and to study the performance of the deep-learning based method under different traffic patterns.

Acknowledgment

We would like to thank the reviewers and editors for their constructive suggestions. This research was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (No. 683253, GraphInt), and the China Fundamental Research Funds for the Central Universities No. 0630/ZK1074, Natural Science Foundation of Fujian Province No. 2018J01105, NSF of China No. U1605254 and No. 61371144.

References

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-scale Machine Learning on Heterogeneous Systems Software available from: tensorflow.org.

Azoff, E.M., 1994. Neural Network Time Series Forecasting of Financial Markets, first ed. John Wiley & Sons, Inc., New York, NY, USA.

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

Bhaumik, S., Chandrabose, S.P., Jataprolu, M.K., Kumar, G., Muralidhar, A., Polakos, P., Srinivasan, V., Woo, T., 2012. CloudIQ: a framework for processing base stations in a data center. In: Proceedings of the 18th Annual International Conference on Mobile Computing and Networking, Mobicom '12. ACM, New York, NY, USA, pp. 125–136.

Box, G.E.P., Jenkins, G.M., Reinsel, G.C., Ljung, G.M., 2015. Time Series Analysis: Forecasting and Control. John Wiley & Sons.

C. M. R. Institute, 2011. C-RAN: the Road toward Green RAN. Tech. rep.. China Mobile Research Institute, Beijing, China.

Checko, A., Christiansen, H.L., Yan, Y., Scolari, L., Kardaras, G., Berger, M.S., Dittmann, L., 2015. Cloud RAN for mobile networks - a technology overview. IEEE Commun. Surv. Tutorials 17 (1), 405–426.

Chen, C., Zhang, D., Li, N., Zhou, Z.-H., 2014. B-planner: planning bidirectional night bus routes using large-scale taxi GPS traces. IEEE Trans. Intell. Transport. Syst. 15 (4), 1451–1465.

Chen, L., Zhang, D., Wang, L., Yang, D., Ma, X., Li, S., Wu, Z., Pan, G., Nguyen, T.-M.-T., Jakubowicz, J., 2016a. Dynamic cluster-based over-demand prediction in bike sharing systems. In: Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp'16. ACM, pp. 841–852.

Chen, L., Zhang, D., Ma, X., Wang, L., Li, S., Wu, Z., Pan, G., 2016b. Container port performance measurement and comparison leveraging ship GPS traces and maritime open data. IEEE Trans. Intell. Transport. Syst. 17 (5), 1227–1242.

Chen, L., Nguyen, T.-M.-T., Pan, G., Jakubowicz, J., Liu, L., Fan, X., Li, J., Wang, C., 2017a. Complementary Base Station Clustering for Cost-effective and Energy-efficient Cloud-ran.
Chen, L., Jakubowicz, J., Yang, D., Zhang, D., Pan, G., 2017b. Fine-grained urban event detection and characterization based on tensor cofactorization. IEEE Trans. Hum. Mach. Syst. 47 (3), 380–391.
Cici, B., Gjoka, M., Markopoulou, A., Butts, C.T., 2015. On the decomposition of cell phone activity patterns and their connection with urban ecology. In: Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc '15. ACM, New York, NY, USA, pp. 317–326.
Cisco, 2016. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper. Tech. rep.. Cisco, San Jose, CA, USA.
Dorffner, G., 1996. Neural networks for time series processing. Neural Netw. World 6, 447–468.
Furno, A., Naboulsi, D., Stanica, R., Fiore, M., 2016. Mobile demand profiling for cellular cognitive networking. IEEE Trans. Mobile Comput. 99, 1.
Gandotra, P., Jha, R.K., 2017. A survey on green communication and security challenges in 5G wireless communication networks. J. Netw. Comput. Appl. 96, 39–61.
Gers, F.A., Eck, D., Schmidhuber, J., 2002. Applying LSTM to time series predictable through time-window approaches. In: Neural Nets WIRN Vietri-01, Perspectives in Neural Computing. Springer, London, pp. 193–200.
Guo, B., Wang, Z., Yu, Z., Wang, Y., Yen, N.Y., Huang, R., Zhou, X., 2015. Mobile crowd sensing and computing: the review of an emerging human-powered sensing paradigm. ACM Comput. Surv. 48 (1), 1–31.
Hamilton, J.D., 1994. Time Series Analysis, vol. 2 Princeton university press Princeton.
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.
I, C.L., Rowell, C., Han, S., Xu, Z., Li, G., Pan, Z., 2014. Toward green and soft: a 5G perspective. IEEE Commun. Mag. 52 (2), 66–73.
J. Research, 2011. Mobile Operator Business Models: Challenges, Opportunities & Adaptive Strategies 2011-2016. Tech. rep.. Juniper Research, New York.
LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521 (7553), 436–444.
Lee, N., Heath, R.W., Morales-Jimenez, D., Lozano, A., 2013. Base station cooperation with dynamic clustering in super-dense cloud-RAN. In: 2013 IEEE Globecom Workshops (GC Wkshps), pp. 784–788.
Li, G.Y., Xu, Z., Xiong, C., Yang, C., Zhang, S., Chen, Y., Xu, S., 2011. Energy-efficient wireless communications: tutorial, survey, and open issues. IEEE Wireless Commun. 18 (6), 28–35.
Lin, J., 1991. Divergence measures based on the Shannon entropy. IEEE Trans. Inf. Theor. 37 (1), 145–151.
Newman, M.E.J., Girvan, M., 2004. Finding and evaluating community structure in networks. Phys. Rev. E 69 (2), 026113.
Nie, L., Jiang, D., Guo, L., Yu, S., 2016. Traffic matrix prediction and estimation based on deep learning in large-scale IP backbone networks. J. Netw. Comput. Appl. 76, 16–22.
Qian, M., Hardjawana, W., Shi, J., Vucetic, B., 2015. Baseband processing units virtualization for cloud radio access networks. IEEE Wireless Commun. Lett. 4 (2), 189–192.
Raghavan, U.N., Albert, R., Kumara, S., 2007. Near linear time algorithm to detect community structures in large-scale networks. Phys. Rev. E 76 (3), 036106.
A. Rajkomar, E. Oren, K. Chen, A. M. Dai, N. Hajaj, P. J. Liu, X. Liu, M. Sun, P. Sundberg, H. Yee, K. Zhang, G. E. Duggan, G. Flores, M. Hardt, J. Irvine, Q. Le, K. Litsch, J. Marcus, A. Mossin, J. Tansuwan, D. Wang, J. Wexler, J. Wilson, D. Ludwig, S. L. Volchenboum, K. Chou, M. Pearson, S. Madabushi, N. H. Shah, A. J. Butte, M. Howell, C. Cui, G. Corrado, J. Dean, Scalable and Accurate Deep Learning for Electronic Health Records, arXiv:1801.07860 [cs]arXiv:1801.07860.
Sang, A., Li, S.-q., 2002. A predictability analysis of network traffic. Comput. Network. 39 (4), 329–345.
Sigwele, T., Alam, A.S., Pillai, P., Hu, Y.F., 2017. Energy-efficient cloud radio access networks by cloud based workload consolidation for 5G. J. Netw. Comput. Appl. 78, 1–8.
Sutskever, I., Vinyals, O., Le, Q.V., 2014. Sequence to sequence learning with neural networks. In: In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems, vol. 27. Curran Associates, Inc., pp. 3104–3112.
Tan, M., Wang, B., Wu, Z., Wang, J., Pan, G., 2016. Weakly supervised metric learning for traffic sign recognition in a LIDAR-equipped vehicle. IEEE Trans. Intell. Transport. Syst. 17 (5), 1415–1427.
Tekinay, S., Jabbari, B., 1991. Handover and channel assignment in mobile cellular networks. IEEE Commun. Mag. 29 (11), 42–46.
Tse, D., Viswanath, P., 2005. Fundamentals of Wireless Communication. Cambridge University Press.
Wang, L., Zhang, D., Wang, Y., Chen, C., Han, X., M'hamed, A., 2016. Sparse mobile crowdsensing: challenges and opportunities. IEEE Commun. Mag. 54 (7), 161–167.
Wang, J., Wang, Y., Zhang, D., Wang, L., Chen, C., Lee, J.W., He, Y., 2017. Real-time and generic queue time estimation based on mobile crowdsensing. Front. Comput. Sci. 11 (1), 49–60.
Yang, D., Zhang, D., Chen, L., Qu, B., 2015. NationTelescope: monitoring and visualizing large-scale collective behavior in LBSNs. J. Netw. Comput. Appl. 55, 170–180.
Zhan, S.C., Niyato, D., 2017. A coalition formation game for Remote radio Head cooperation in cloud radio access network. IEEE Trans. Veh. Technol. 66 (2), 1723–1738.
Zhang, G.P., 2003. Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing 50 (Suppl. C), 159–175.
Zhang, G.P., Qi, M., 2005. Neural network forecasting for seasonal and trend time series. Eur. J. Oper. Res. 160 (2), 501–514.
Zhang, D., Guo, B., Yu, Z., 2011. The emergence of social and community intelligence. Computer 44 (7), 21–28.
Zhang, J., Ji, Y., Xu, X., Li, H., Zhao, Y., Zhang, J., 2016. Energy efficient Baseband unit aggregation in cloud radio and optical access networks. J. Opt. Commun. Netw. 8 (11), 893–901.
Zhang, Y., Song, B., Zhang, P., 2017. Social behavior study under pervasive social networking based on decentralized deep reinforcement learning. J. Netw. Comput. Appl. 86, 72–81 special Issue on Pervasive Social Networking.
Zhao, Z., Peng, M., Ding, Z., Wang, W., Poor, H.V., 2016. Cluster content caching: an energy-efficient approach to improve quality of service in cloud radio access networks. IEEE J. Sel. Area. Commun. 34 (5), 1207–1221.
Zheng, K., Yang, Z., Zhang, K., Chatzimisios, P., Yang, K., Xiang, W., 2016. Big data-driven optimization for mobile networks toward 5G. IEEE Netw. 30 (1), 44–51.

Longbiao Chen is an assistant professor with Fujian Key Laboratory of Sensing and Computing for Smart City, Xiamen University, China. He received his Ph.D. degree in computer science from Zhejiang University in 2016, under a joint-cultivated doctoral program with University of Paris VI, France. His research interests are mobile computing, urban data mining, and ubiquitous computing. Dr. Chen has published over 20 papers in toptier journals and conferences, and received the 2015 and 2016 UBICOMP Honorable Mention Awards.

Dingqi Yang is a senior researcher in the Department of Computer Science, University of Fribourg, Switzerland. He received his Ph.D. in Computer Science from Pierre and Marie Curie University (Paris VI) and Institut Mines-TELECOM/TELECOM SudParis, where he won both the Doctorate Award and the Institut Mines-TELECOM Press Mention in 2015. His research interests lie in big social media data analytics, ubiquitous computing and smart city applications.

Daqing Zhang is a Full Professor at Telecom SudParis, Institut Mines- Telecom, France. His research interests include context-aware computing, urban computing, mobile computing, big data analytics, pervasive elderly care, etc. Dr. Zhang has published more than 200 technical papers in leading conferences and journals, where his work on context model is widely accepted by the pervasive computing, mobile computing and service-oriented computing communities. Daqing Zhang obtained his Ph.D. from University of Rome "La Sapienza" in 1996.

Cheng Wang received the Ph.D. degree in information and communication engineering from the National University of Defense Technology, Changsha, China, in 2002. He is currently a Professor with and the Associate Dean of the School of Information Science and Technology, Xiamen University, Xiamen, China. He has authored more than 80 papers. His research interests include remote sensing image processing, mobile LiDAR data analysis, and multisensor fusion.

Jonathan Li received the Ph.D. degree in geomatics engineering from the University of Cape Town, Cape Town, South Africa. He is currently a Professor with the School of Information Science and Engineering, Xiamen University, Xiamen, China. He is also a Professor with and the Head of the GeoSTARS Lab, Faculty of Environment, University of Waterloo, Waterloo, ON, Canada. He has coauthored more than 300 publications, more than 100 of which were published in refereed journals. His current research interests include mobile computing, communication networks, and information extraction from mobile LiDAR point clouds.

Thi-Mai-Trang Nguyen is an associate professor at University Pierre and Marie Curie (Paris 6) and doing research at Laboratoire d'Informatique de Paris 6 (LIP6), France. She received the PhD Degree in Computer Science from University of Paris 6, France, in 2003. The PhD thesis was cosupervised and carried-out at Ecole Nationale Superieure des Telecommunications (ENST-Paris). From 2004 to 2006, She was postdoctoral researcher at France Telecom in Rennes, France and at University of Lausanne, Switzerland. Her research interests include network architecture, network protocol design, and network data analytics.