Journal of Ambient Intelligence and Humanized Computing (2021) 12:8229–8240
https://doi.org/10.1007/s12652-020-02557-z

ORIGINAL RESEARCH

Saving bandwidth and energy of mobile and IoT devices with link predictions

Gabriel Orsini[1] · **Wolf Posdorfer**[1] · **Winfried Lamersdorf**[1]

Received: 13 April 2020 / Accepted: 15 September 2020 / Published online: 28 September 2020

Abstract
Use cases in the Internet of Things (IoT) and in mobile clouds often require the interaction of one or more mobile devices with their infrastructure to provide users with services. Ideally, this interaction is based on a reliable connection between the communicating devices, which is often not the case. Since most use cases do not adequately address this issue, service quality is often compromised. Aimed to address this issue, this paper proposes a novel approach to forecast the connectivity and bandwidth of mobile devices by applying machine learning to the context data recorded by the various sensors of the mobile device. This concept, designed as a microservice, has been implemented in the mobile middleware *CloudAware*, a system software infrastructure for mobile cloud computing that integrates easily with mobile operating systems, such as Android. We evaluated our approach with real sensor data and showed how to enable mobile devices in the IoT to make assumptions about their future connectivity, allowing for intelligent and distributed decision making on the mobile edge of the network.

Keywords Mobile clouds · Internet of Things · Context awareness · Context forecast

1 Introduction

Mobile devices such as smartphones, wearables and sensor nodes have become more powerful every year. Nevertheless, they often rely on the resource augmentation through centralized resources, enabling a multitude of cloud-augmented mobile applications. Examples are location-based advertising, real-time sensor networks, the *Nvidia Shield* videogaming console (Nvidia 2019), which computes parts of the gameplay on remote resources, or the voice recognition assistant *Siri* (Apple 2019). Common to these use cases is the fact, that they rely on a preferably fast and stable connection to centralized or edge clouds (Abbas et al. 2018). However, the typical scenario of a moving user illustrated in Figure 1 shows that this is often not the case. Even with upcoming 5G networks it is assumed that the obstacle of the intermittent connectivity of mobile devices will persist (Patel et al. 2017).

Knowledge about the future connectivity of mobile devices allows developers to design their applications accordingly and allow an improved usability and user experience, for example by deciding whether to prefetch data or postpone the synchronization with cloud services. Moreover, information about the current and future bandwidth can be used to decide when to activate the wireless network interfaces, e.g. GSM or WiFi, of the mobile device. Hereby, the interfaces will only be activated in situations where a high bandwidth can be achieved, improving the ratio between the required energy and the transferred data and thus saving energy, often the most limited resource on mobile devices.

Currently, many of the proposed solutions in the domain of connectivity forecasts are limited to specific use cases and only allow short-term forecasts. Filling this gap, this paper, which is based on our previous works in the domain of context forecasts (Orsini et al. 2016, 2019), aims to provide a concept for the connectivity forecast on a multitude of different mobile devices, allowing them to reason about their future bandwidth. This way, a more efficient interaction between mobile devices as well as between cloud and edge cloud resources becomes possible and can provide a

✉ Gabriel Orsini
orsini@informatik.uni-hamburg.de

Wolf Posdorfer
posdorfer@informatik.uni-hamburg.de

Winfried Lamersdorf
lamersd@informatik.uni-hamburg.de

[1] Distributed Systems Group, Department of Computer Science, University of Hamburg, Vogt-Koelln-Str. 30, 22527 Hamburg, Germany

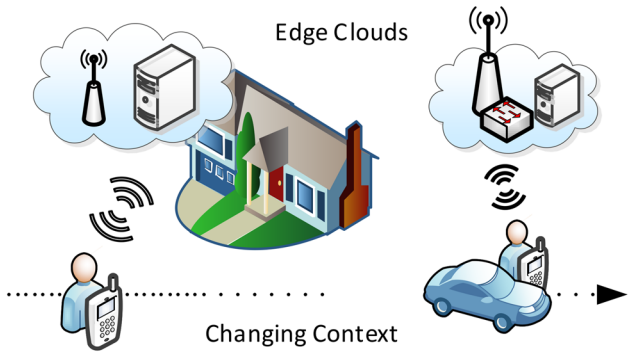

Fig. 1 Mobile device moving between edge clouds

higher user experience while at the same time saving energy. Hence, the contributions of this paper can be summarized as follows:

- The concept of a modular connectivity service that is able to run on a multitude of devices. Firstly through the dynamic adaptation of the computational requirements and secondly through the adaptation to the relevant sources of information.
- Insights into specific sub-problems of this concept, such as the cold start problem and a simulation that determines, which and how much information about the context[1] of the mobile device is required to allow reliable forecasts.
- A quantitative evaluation of the presented approach based on realistic mobile device usage data provided by the Lausanne Data Challenge Campaign (LDDC) (Laurila et al. 2013).

The remainder of this paper is structured as follows: Sect. 2 summarizes related work, afterwards Sect. 3 derives requirements and describes our own approach, that is subsequently evaluated in Sect. 4. At the end, we summarize our findings, highlight open challenges and give prospects for future work in Sect. 5.

2 Related work

As summarized in (Orsini et al. 2018a), the task of forecasting the future connectivity of a mobile device can either be seen as a software engineering- or a networking problem. Seen primarily as a software engineering problem, solutions in the domain of mobile cloud computing like *Serendipity* (Shi et al. 2012) try to distribute computation tasks among other nearby mobile devices to speed up computation or to save energy. Hereby, Serendipity takes into account the future state of the mobile network connection by forecasting its reliability. Similarly, *IC-Cloud* (Shi et al. 2013) focuses on the challenge of dynamically offloading computation tasks to cloud resources, by taking into consideration that the necessary code and data can be delivered and the results received in time before the next link failure is likely to happen. Specifically developed for the application in mobile edge computing, Sato et al. (Sato and Fujii 2017) recently proposed a radio environment aware algorithm that is able to forecast the mobile connectivity between mobile nodes in access points.

Seen primarily as a networking problem, solutions like *BreadCrumbs* (Nicholson and Noble 2008) try to forecast the future connectivity to WiFi hotspots based on a model of the environment. Recorded sensor data is used to generate user-based models which are then applied to schedule the network usage based on connectivity forecasts. BreadCrumbs relies on the fingerprinting of hotspots that is combined with GPS data to forecast a mobile user's bandwidth. Focusing on the aspect of the location forecasting even further, in *NextPlace* (Scellato et al. 2011) a non-linear method is employed to forecast the time and duration of a user's next visit to one of his significant places. Their method identifies patterns in a user's mobility history that are similar to his recent movements in order to forecast his behavior. Similarly, Anagnostopopulus et al. (Anagnostopoulos et al. 2011) employ supervised learning to perform a classification of trajectories which is then used to forecast the future location of mobile users.

Further related works can either be found in the domain of mobility- and connectivity forecasts such as in mobile ad-hoc networking (MANET) or vehicular ad-hoc networking (VANET) (Fernando et al. 2013; Shiraz and Gani 2014; Lee 2008). Summarizing the previous findings, it can be concluded that several solutions have been proposed to contribute to the problem of connectivity forecasting. However, current solutions are either not able to operate on a broad range of mobile devices or are not able to provide the level of accuracy, required in IoT scenarios such as computation offloading (Orsini et al. 2018a).

3 Bandwidth forecast service

In this chapter, we will first define the design goals for the bandwidth forecast before we discuss different architectures along with the integration of forecasts into mobile applications. Subsequently, we choose different models that we expect to fit the problem of the bandwidth forecast.

[1] According to Dey and Abowd, context is any information that can be used to characterize the situation of an entity. For details see (Dey and Abowd 1999).

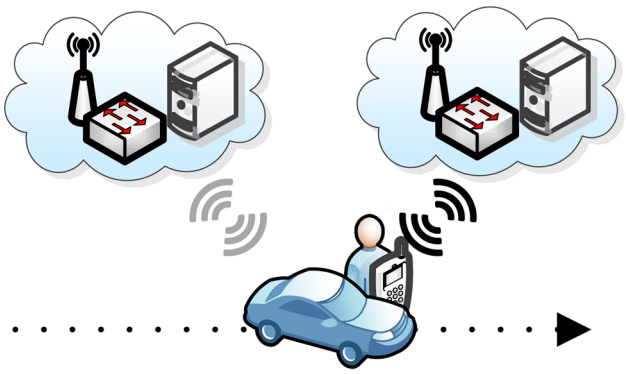

Fig. 2 Changing and intermittent connectivity between access points

Afterwards, we choose the dataset to perform our analysis on and decide how to prepare the data.

3.1 Problem statement

As discussed in Sect. 1, forecasting the connectivity of a mobile device allows a more effective use of its resources. In particular, this can help to make more efficient use of the limited bandwidth and hereby also the available energy. To achieve this goal, mobile applications need to be provided with the information about the current and future bandwidth to be able to decide when to use that resource. For example, when to make extensive use by prefetching data or when to conserve this resource by postponing the synchronization with cloud services.

Accordingly, a bandwidth forecast is required for users that are moving between access points such as GSM base stations and WiFi hotspots, as illustrated in Fig. 2.

3.2 Design goals

Apart from the functional requirement of a forecast with high accuracy, developing the *Bandwidth Forecast Service* on a mobile device requires to meet several non-functional requirements. Accordingly, we summarize the specific design goals for the development of this service with the following key requirements:

- *Support for different time intervals and forecast horizons* Apart from real-time applications, it is often sufficient to forecast the average bandwidth in a certain time interval. The duration of the time interval depends on the use case and can range between minutes and hours. Likewise, the required forecast horizon for the service ranges from the next minute to a bandwidth forecast for the next day.
- *Resource-conserving and customizable* The service dynamically selects the appropriate learning algorithms based on the required accuracy and the resources available on the mobile device.
- *Can handle small amounts of data:* The service is able to operate even when there is little data available on the mobile device.
- *Privacy aware* It is possible to process all information on the mobile device itself. The requirement for external data processing should be optional.
- *Open for extension* The service is open for extension with new learning algorithms or new data sources.

3.3 Architecture

In order to forecast the bandwidth of a mobile device that is subject to changing connectivity, it is necessary to generate the forecast on the affected device itself, to allow for continuous availability of the service. Thus, although the use of a forecast model, has to be done on the mobile device, the creation of this model, the training, does not necessarily have to be performed on the mobile device as well. This leads to a number of options for how the forecasting model is generated and what data is used for it.

For the generation of the forecasting model, the following alternatives can be envisioned:

- *Local training* If the training data contains sensitive information, it may be necessary to carry out the training on the mobile device itself.
- *Remote training* Depending on the performance of the mobile device, it may be necessary to transfer the recorded context data into a more powerful infrastructure. In addition, a longer history of this data can be held in this infrastructure, which can be beneficial for the quality of the learned model.

Based on the ability to offload the training of the model into a more powerful infrastructure, not only the data of a user, but the data of many users can be used to train the model. Following the idea of distributed machine learning approaches Google recently proposed a system design (Bonawitz et al. 2019) that employs the concept of federated learning (McMahan et al. 2016; Rahman and Rahmani 2018). Tailored to the domain of mobile devices, federated learning establishes a distributed intelligence by using the recorded context data to train forecast models on the respective mobile device. These models are then transferred and aggregated into a global model, which is then distributed to all participating devices, as shown in Fig. 3 (left).

Especially, when there is not enough data available, this approach can be highly beneficial to train a suitable forecast model. Furthermore, this approach is well suited to problems where the use case and the nature of the data is the same amongst all of the participating devices. In this case, large

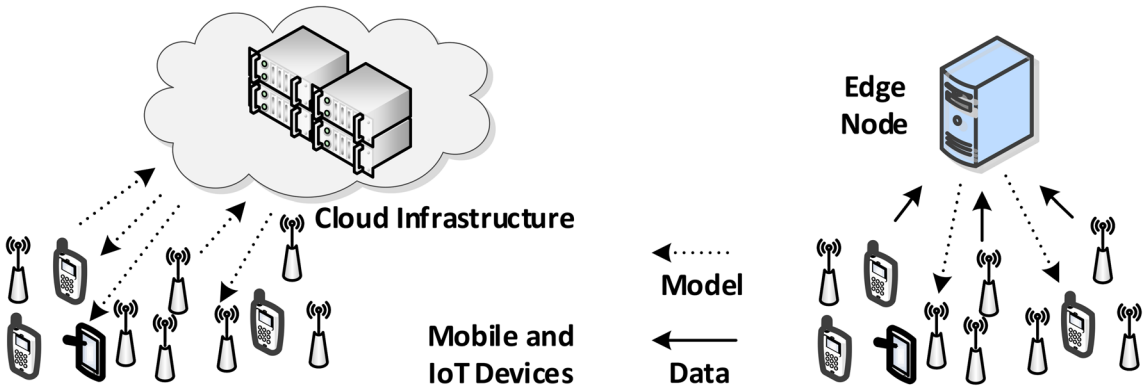

Fig. 3 Distributed learning: federated learning (left) and edge-centric learning (right)

amounts of data generated by many users can help to train a global model.

However if a high level of heterogeneity is present amongst the participating devices, which is the case in particular in the IoT and, in addition, a part of the participating devices is not able to carry out the computation-intensive training of a forecast model itself, this approach does not appear promising. Accordingly, we decide to adapt this approach, calling it edge-centric computing, illustrated in Fig. 3 (right). It works as follows: Since many of these devices are already transmitting data, it is reasonable to consolidate this data on a sufficiently powerful device and to train a model there. This also has the advantage that the resulting model is automatically optimized for the context of the edge cloud in which it is used.

3.4 Embedding forecasts into mobile applications

As proposed in the problem statement earlier, providing a mobile application with forecasts can help to make more efficient use of the available resources. But integrating such of a forecast raises two important questions:

- How can the forecast be woven into the mobile application's business logic?
- What type of forecast model is suitable for a bandwidth forecast?

To be easily integrated, the consideration of the forecasts should follow the same way the application's business logic is implemented. Accordingly the future bandwidth, which is typically the output of a regressor, needs to be reduced to a simple categorical or numerical variable, reflecting the available amount of bandwidth. Based on this universal approach, it can often be more useful to adjust the point forecast by making use of the underlying probability distribution. This way, a confidence interval can be used to forecast the minimum as well as the maximum bandwidth that can be assumed with high confidence. Another typical approach might be to reduce the bandwidth forecast to a classification problem, resulting in a forecast about whether there will be a connection or not, extended by an adjustment of the forecast model, depending on which misclassification is less desirable. Moreover, there might be specific use cases that benefit from other types of forecasts. Nevertheless, the most suitable type of forecast often highly depends on the use case, which is why the average expected bandwidth often depicts a good starting point.

3.5 Model selection

Various approaches have been proposed to forecast the future context of mobile users and their devices. For a general overview, we refer to surveys such as (Sigg 2008) and (Mayrhofer 2004). Subsequently, in this work we focus on a promising pre-selection based on the aforementioned surveys as well as the related works in Sect. 2, that have presented promising approaches to identify patterns in contextual data and are able to translate those relationships into a forecast. Accordingly, during the selection of suitable algorithms we focus on classification and regression techniques that have been successfully used in the domain of context forecasts.

In (Lim and Dey 2010) the authors have surveyed which machine learning algorithms are commonly used in scenarios where context data needs to be forecasted. This evaluation was refined in (Perera et al. 2014) and the following models were found to be the ones that where most often used:

- Decision trees (15 %)
- Rule-based systems (54 %)
- Hidden Markov models (13 %)
- Naive Bayes (13 %)
- Support vector machines (4 %)
- k-nearest neighbor (2 %)

Based on the requirements identified in Sects. 3.2 and 3.4, we prefer model-based learners over instance-based learners due to their lower storage requirements. Furthermore, we extend our selection to robust forecast models from the domain of time-series forecasts, that we expect to perform equally well (Hyndman and Athanasopoulos 2018).

Accordingly, as an extension to the algorithms used in the related works' as well as a baseline for the evaluation, a naive forecasting model is chosen, that assumes the current observation to persist in the future (NAIVE).

Furthermore, as a second baseline as well as a simple model for mobile devices with very limited resources, we choose an autoregressive model (AR) that tries to forecast the future bandwidth by just taking into account past observations of the forecasting target itself. The *AR(p)* model assumes a linear dependency on its own previous values, the constant term c, the intercept, as well as the stochastic term ε. Equation 1 defines the AR model as follows:

$$y_t = c + \sum_{i=1}^{p} \beta_i y_{t-i} + \varepsilon_t \tag{1}$$

Based on the AR model, a more sophisticated approach appears to include more information than just the target variable itself.

As we assume that arbitrary additional sensor data can be highly multicollinear, we require a type of regression estimator that is well-suited to deal with such types of issues. Least Absolute Shrinkage and Selection Operator (LASSO) (Tibshirani 1996) is an estimator that performs an L1 regularization which adds a penalty equal to the absolute value of the magnitude of coefficients encouraging simple, sparse models, i.e. models with fewer parameters. This approach leads to the optimization problem shown in Eq. 2, where λ is a nonnegative regularization parameter that can be either tuned manually or using a grid-search and p denotes the number of features.

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \frac{1}{N} \sum_{n=1}^{N} \left(y_n - \beta x_n\right)^2 + \lambda \sum_{i=1}^{p} |\beta_i| \tag{2}$$

Solving this optimization problem requires substantially more computation power compared to the training of the AR model, but promises to better capture the relationship between the mobile device's context and its future bandwidth.

In line with the previous selection of promising machine learning algorithms, we choose decision trees to cover higher order interactions between the individual variables of the sensor data and the target variable. Using gradient boosting (Friedman 2000), which uses ensembles of weak forecast models to iteratively build a stronger model, we aim to build a model that has a low bias and low variance at the same time. As an implementation, we choose *XGBoost* (v. 0.8.2) (Chen and Guestrin 2016) that currently is considered one of the state-of-the-art machine learning algorithms to deal with structured data (Nielsen 2016).

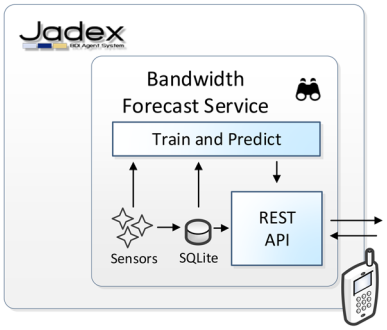

Fig. 4 Forecasting service architecture

Finally, it should be noted that all of the selected learners are model-based, meaning it is not necessary to keep the data that has been used in the training phase and highly reduces the memory footprint of the presented approach. Also, this allows to eliminate many privacy concerns, in case the model is used as part of a distributed approach and thus is shared amongst other devices, as explained in Sect. 3.3.

3.6 Runtime environment

As illustrated in Fig. 4, we employ standard Java and Android technology to implement the Bandwidth Forecast Service as a microservice which is integrated into the CloudAware mobile middleware presented in (Orsini et al. 2018a).

CloudAware is based on the *Jadex* (Pokahr and Braubach 2013) middleware that provides infrastructure components, such as service discovery in mobile environments, and allows to expose the microservice. The microservice itself has been implemented to only be activated when a forecast is actually requested, mainly to save energy. Furthermore, an accuracy-parameter can be sent alongside with the request to save energy as a less accurate forecast model is used then.

To ensure effective bandwidth forecasting for mobile applications, it is necessary to implement the computationally intensive forecast as efficient as possible. Accordingly, special attention should be paid to the implementation of the selected learning algorithms. For this purpose, the runtimes of a prediction on a current smartphone have been evaluated and are summarized in Table 1. If also the training is carried out locally, the corresponding runtimes also become relevant and are hence shown in the subsequent columns.

The measurements indicate, that the predictions are rather resource-efficient. Furthermore, the time required to generate a forecast is negligible, regardless of which algorithm is

Table 1 Computation times for training and prediction

Runtime (s)	Prediction	Training		
Samples	1	5346	31,252	155,218
AR	6.2E–04	0.2	0.4	0.4
LASSO	7.5E–04	2.8	7.1	3.6
XGB	1.02E–01	479.0	2393.2	2900.6

used. Equally, the time required to train the models, using 102 features and 500 decision trees for the XGB algorithm, suggests that also the training can be performed locally, assuming the performance of a current smartphone. Even devices with less computational power available are able to carry out the training of at least some models, taking into account the short training times of the AR and LASSO learning algorithms.

3.7 Datasset and data preparation

In January 2009 the Nokia Research Center Lausanne, the Idiap Research Institute and the École polytechnique fédérale de Lausanne initiated the creation of a large-scale mobile data research.

This included the design and implementation of the LDDC, an initiative to collect sensor data from smartphones created by almost 200 volunteers in the Lake Geneva region over a period of 18 months (Laurila et al. 2013). According to (Crawdad 2019) it is still the largest dataset that contains information about mobile devices' bandwidth and sensor data, which is why we chose the LDDC dataset to derive the following information as the input to our simulation:

- *Connectivity and bandwidth* GSM/WiFi/ Bluetooth state (on/off), discovered MAC addresses and GSM cells, signal strength of WiFi as well as GSM cells, extended with our own measurements to get an assumption on the available bandwidth.
- *General information about the mobile device itself* time since the last user interaction, silent mode state, charging state, remaining energy, free memory.
- *Date, time and location* calendar events, average and estimated remaining duration of stay at the current location.
- *Reasoned attributes* estimated duration of stay at the same WiFi access point or GSM cell, user is at home/ work, traveling, moving, or resting.

This information has been transformed into panel data containing observations over a period of at least 18 months per user. Hereby, we used different time intervals of 2, 10 and 60 minutes to be able to forecast the bandwidth in different granularities.

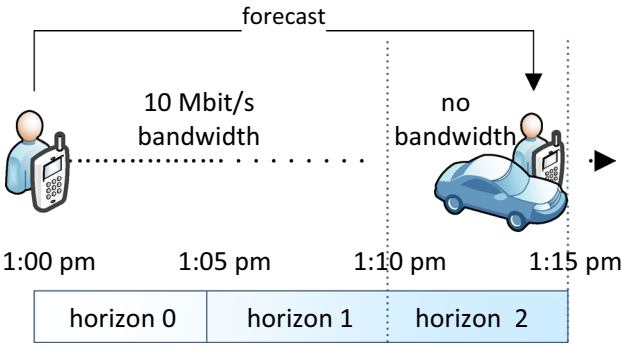

Fig. 5 Operation of the forecasting service: interval-based forecasts for different forecast horizons

4 Evaluation

We first evaluate the performance of the selected algorithms. Afterwards, we extend the evaluation and focus on the amount and type of data that is required to achieve good forecasting accuracy. Subsequently, we evaluate the performance of generalized models, that can be used in case there is no data available at all for a specific user.

4.1 Simulation setup and goals

The developed Bandwidth Forecast Service is evaluated by selecting 20 users who have provided data of at least 18 months to the LDDC dataset. To evaluate the performance we simulate the usage of a mobile device, as described in (Orsini et al. 2018b). This model uses the context data from the LDDC dataset to simulate a mobile device in its constantly changing environment, which aims to reflect the real-world usage throughout the whole period of the observations. Hereby we aim to forecast the future bandwidth in a defined future time interval, as illustrated in Fig. 5. In line with the research goals defined in Sect. 1, we focus on the following three key aspects:

- Which context information correlates most with the future bandwidth of a mobile device and should be used to train a model?
- Which of the selected models performs best in capturing the relationships and can their respective weaknesses be alleviated through a combination of models?
- How much data is required and are we able to generalize patterns across individual users to form a distributed intelligence?

4.2 Feature importance

To answer the question which context variable supports the forecast of a future bandwidth the best, we analyze their

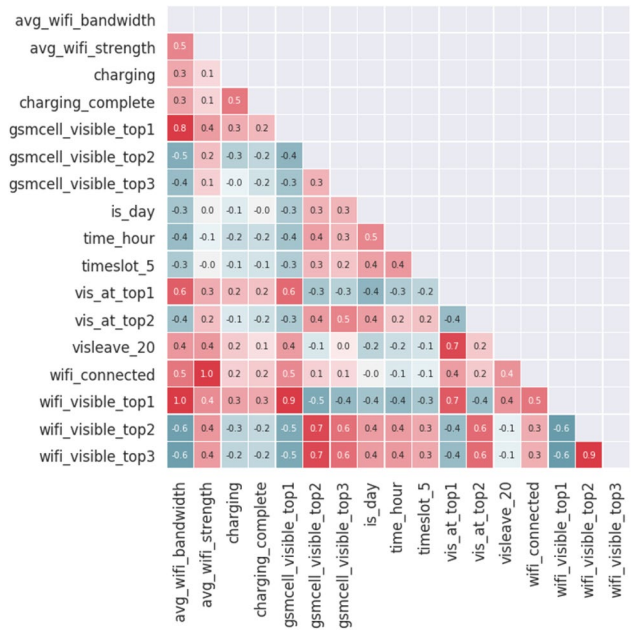

Fig. 6 Correlation matrix for the context variables

linear relationships using the Pearson correlation coefficient as shown in Fig. 6. Due to the lack of space, we just show the most relevant correlations of the total 112 features contained in the panel. As expected, the current bandwidth, exemplified as *avg. wifi bandwidth*, is primarily correlated with context variables that describe the current connectivity of the mobile device as well as its physical location. Although the correlation plot in Fig. 6 shows a high correlation between many of the context variables, it supports the hypothesis about the relevance of physical and temporal dependencies.

Summarizing the first aspect of the evaluation, it can be concluded that the availability of these variables helps to forecast the future bandwidth of a mobile device. However, this assumption mainly refers to moving devices that users carry with them. Other scenarios in the IoT, apart from the bandwidth forecast, may exhibit other relationships and hence other variables can become the most relevant features for the forecast.

4.3 Forecasting accuracy

The aforementioned simulation is carried out for 20 users of the dataset to validate the general applicability of our approach. For some parts of this evaluation more meaningful results can be shown by looking at specific users or timeslices. Therefore, some of the following evaluations are shown only for distinct subsets of the entire simulation results, as mentioned in the captions of the respective figures.

Accordingly, we perform an ex-post evaluation using a rolling window approach for which we assume a weekly retraining of the models. This approach takes into account that at the beginning of the simulation period only little data is available. Where applicable, the hyperparameters are tuned using grid searches and cross-validation. Although this task would typically not be carried out in a real-world implementation, it helps to estimate the full potential of the evaluated models.

To show the general usefulness of the trained models, Fig. 7 presents the actual and the forecasted bandwidth for different context intervals in conjunction with a fixed horizon. Fig. 8 shows different forecasting horizons together with a fixed context interval, both on a randomly chosen day of the simulation.

A first look at the data leads to the assumption that only the XGBoost model is able to properly capture the relationships between the context data and provides the most accurate forecasts of the future bandwidth of a mobile device. Moreover, Fig. 9 presents the corresponding error distributions, underlining that XGBoost also provides the most unbiased forecasts. However, for a proper evaluation of the forecasting accuracy, an appropriate error-measure needs to be selected. Since this depends heavily on the use case, we choose the typical linear and quadratic error measures, shown in Table 2 for a context interval of one hour.

With a root mean squared error (RMSE) ranging from 38.7 for a bandwidth forecast for the next hour to a RMSE of 54.3 for a forecast in 20 h, XGBoost appears to be the most robust and accurate predictor for this problem. Nevertheless, the LASSO and the AR model can depict interesting alternatives, when computation resources are low or the use case does not require such a high level of accuracy.

The results of the simulation also show that each of the chosen models has strengths and weaknesses with respect to the desired error measure and forecasting horizon. This suggests the usage of a combination of the existing models, hereafter referred to as ”COMBINED”. This model uses an XGBoost model as an ensemble learner that combines the forecasts of the existing four models, employing the forecast horizon as an additional feature. As depicted in Fig. 10 and Table 2, the error distribution of this model shows, that it is able to provide the most accurate and unbiased forecasts among all selected models.

Summarizing the second aspect of the evaluation, Fig. 11 shows the distribution of the RMSE for the different users. Here, it can be seen that, depending on the specific usage patterns, a certain deviation of the mean errors can be expected. Therefore, depending on the use case, the benefit of a highly uncertain forecast needs to be assessed before it is integrated into a mobile application.

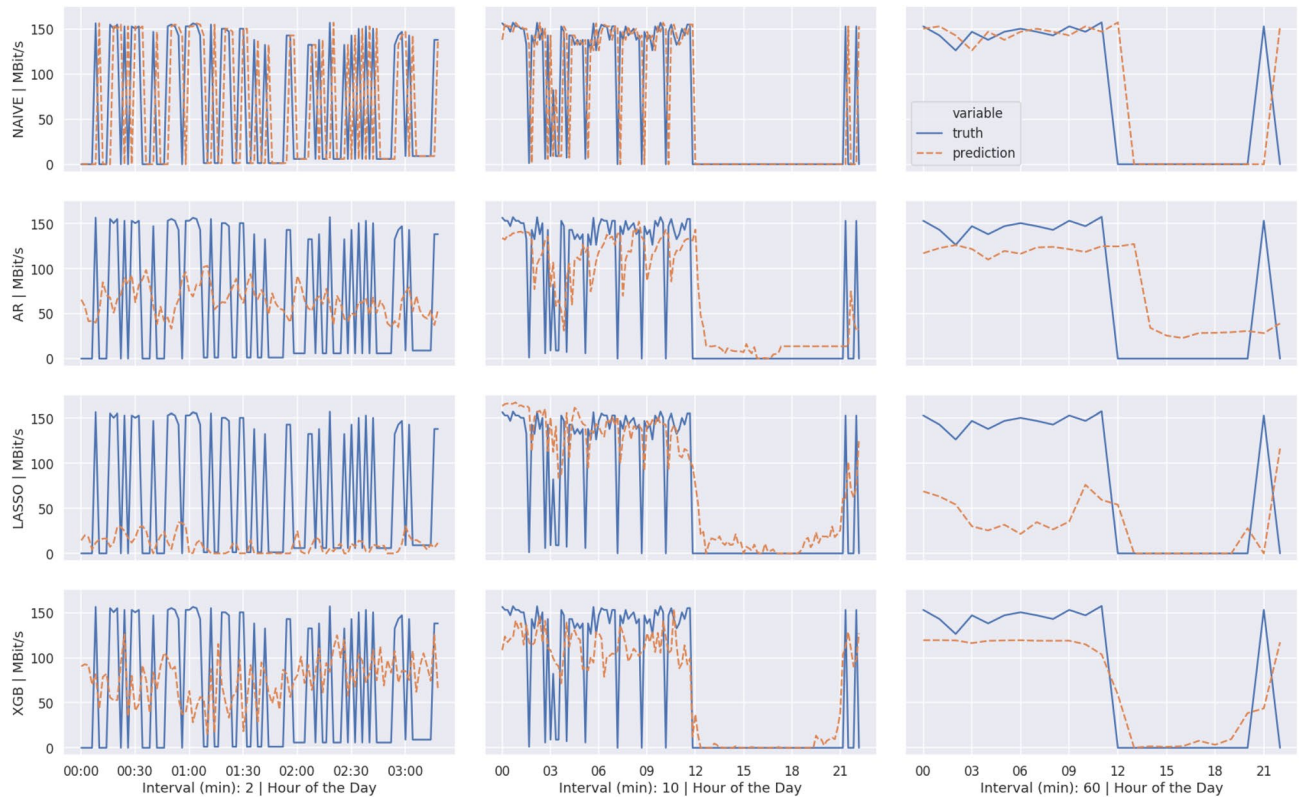

Fig. 7 True and forecasted bandwidth for different models and context intervals, horizon: next interval

4.4 Selecting training data

Next to the selection of an appropriate model, choosing the right data to train a forecast model is also a crucial task, especially in scenarios where the usage, the physical location or the surrounding environment can lead to changes in the network topology at any time, either temporarily or indefinitely.

If a forecast model is trained without the majority of the expected events, it is likely that it fails to map its generalized relationships to new events adequately. In practice, this means that if a mobile device on which a forecast model was trained was not exposed to a particular situation, even a very good forecasting model will not or not adequately forecast that event. At the same time, if too many old events are present in the training set and their temporal order is not taken into account, outdated patterns are learned, and the adaptation to the current environment of a mobile device will be impaired.

Both cases often affect the forecasting quality and hence, must be prevented. Accordingly, we evaluated the forecasting quality using different sizes of training sets, addressing the third aspect of the previously defined goals of the evaluation: what data should be taken into account while building a forecast model. Although this statement is based solely on the use of the LDDC dataset, it can be seen in Fig. 12 that with approximately 500 observations, a good forecasting quality can be reached, with respect to the achievable optimum for this model. Depending on the selected model, it can also be seen that larger amounts of training data, i.e. more than 10,000, can have a negative impact on the forecasting accuracy, as outdated patterns might be overweighted to the disadvantage of current, relevant ones.

4.5 Generalized models and combined forecasts

Based on the findings of the previous two subsections, we continue to address the last aspect of the evaluation. We analyze if a generalized model, trained with the data of other users and their mobile devices is able to serve as a generalized model for devices on which no training data is available or on which the training of a model is not possible due to their limited resources. Accordingly we use the sensor

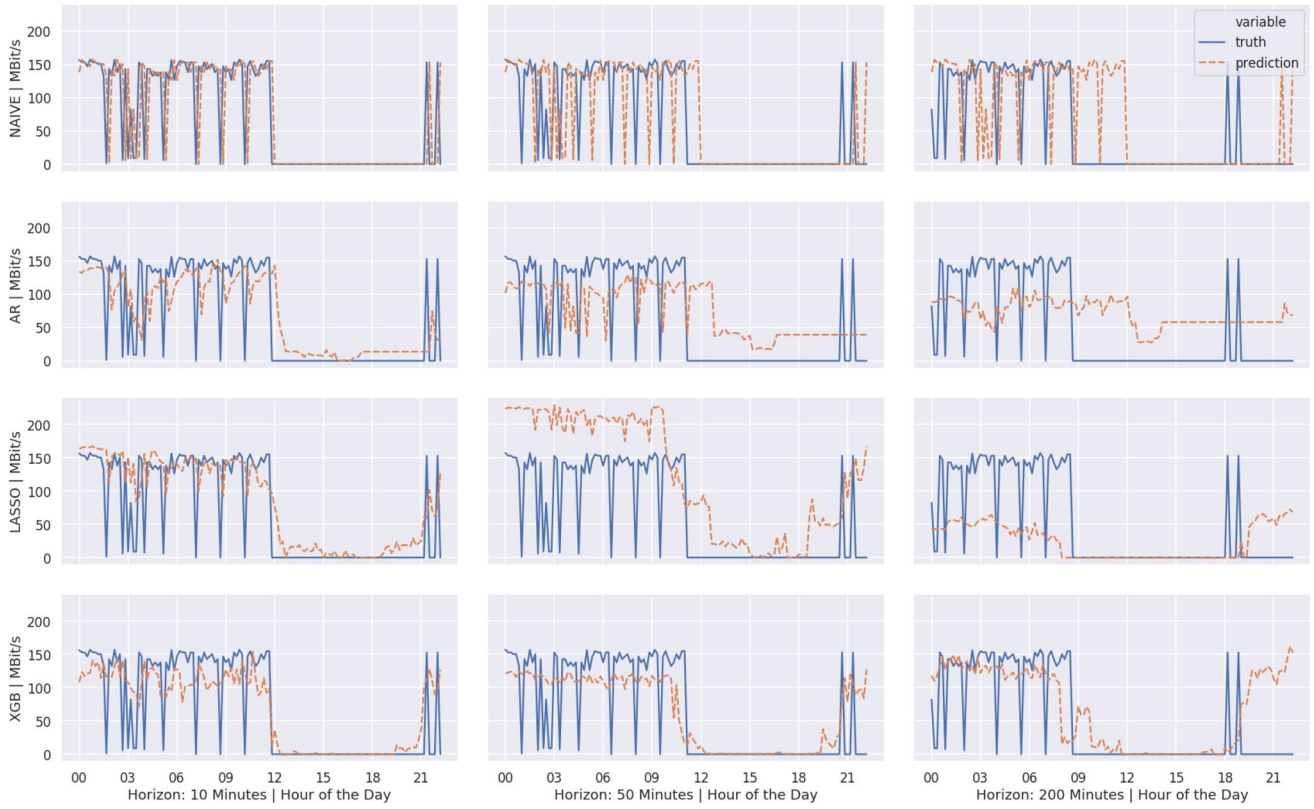

Fig. 8 True and forecasted bandwidth for different models and horizons, interval: 10 minutes

data of ten users to forecast the connectivity of other ten users. As shown in Fig. 13, it can be seen that the relevant amount of data to train an individual model is available after three to four weeks. Using such a generalized model also helps to mitigate the cold start-problem. Switching from the pre-trained model to a user-specific model can easily be decided by measuring the ex-post accuracy and switching accordingly.

5 Conclusion

In this paper, we proposed a novel approach for a Bandwidth Forecast Service for mobile devices in mobile edge and IoT scenarios.

Following a survey of related works, we pointed out the design goals and discussed existing concepts in order to design our own approach of a forecasting service. Subsequently, using real usage data of the LDDC, we highlighted the benefit of using the mobile devices' sensor data to forecast their future bandwidth. Here, we showed that the XGBoost model is best suited to forecast the future connectivity of mobile devices in a constantly changing environment, only surpassed by a combination of models. Afterwards, we highlighted which sensor data is the most relevant for this task and discussed alternatives, in case this data is not available. Although XGBoost was able to provide a high forecast accuracy, its training phase might need to be offloaded to more powerful cloud resources. We addressed this issue with a federated approach that allows to provision less powerful devices with pre-trained models of the same edge cloud or from a larger amount of users, forming a distributed intelligence.

This contribution can support a multitude of scenarios where the limited bandwidth of a mobile device needs to be forecasted. Mainly to mitigate the effect of upcoming network bottlenecks by either delaying, advancing or adapting data transfers, but also to save energy. Furthermore, the discovered relationships can also positively influence other scenarios that require to forecast the context of mobile devices.

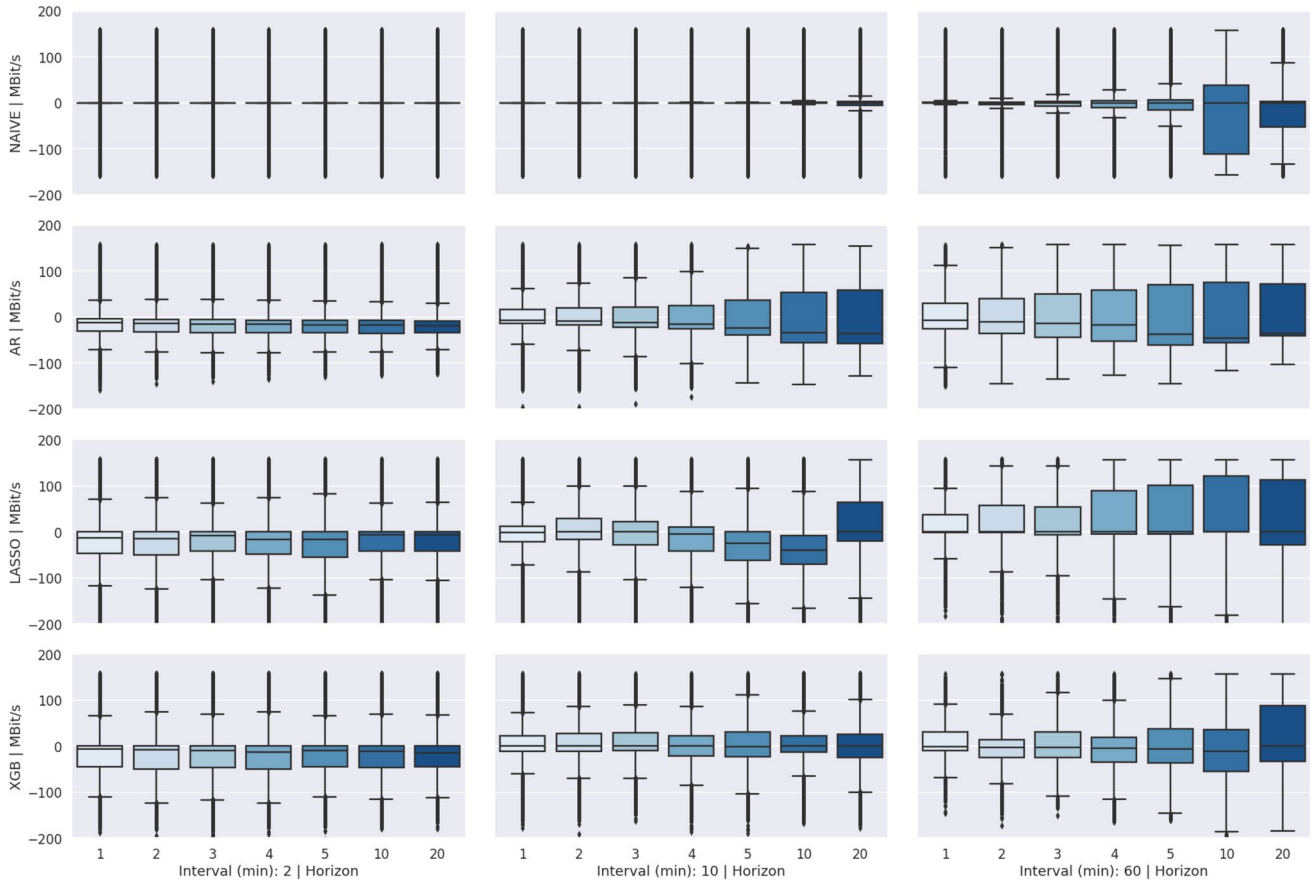

Fig. 9 Error distributions for different models, context intervals and horizons

Table 2 Error measures for different models and horizons

Error	Mean Abs. error (MAE)			Mean Abs. Perc. error (MAPE)			Root mean squared error (RMSE)		
Horizon	1	5	20	1	5	20	1	5	20
AR	37.2	46.5	44.7	85.5	118.3	124.7	48.4	55.1	54.0
COMBINED	18.6	19.1	18.9	44.4	51.6	54.6	29.5	30.8	31.2
LASSO	33.6	50.6	55.6	77.4	131.2	164.7	47.8	68.6	75.4
NAIVE	20.8	41.4	47.6	49.6	107.9	133.0	43.5	65.4	71.2
XGB	26.9	35.9	40.6	62.0	92.2	110.6	38.7	48.7	54.3

5.1 Limitations and future work

Nevertheless, to reliably forecast the wireless connectivity by forecasting the users mobility patterns is a complex task and still considered an open challenge (Farris et al. 2018; Patel et al. 2017). In our future research, we plan to optimize the bandwidth forecast by only choosing the most relevant features and extend the evaluation to simulate different use cases that make use of the bandwidth forecast. Hereby, we aim to estimate the actual benefit of our approach. Furthermore, the selection of the examined models is the result of a preselection. In our future work, we will explore other categories of models, tailored to the quickly changing context as well as to the limited computation and memory resources of mobile and IoT devices.

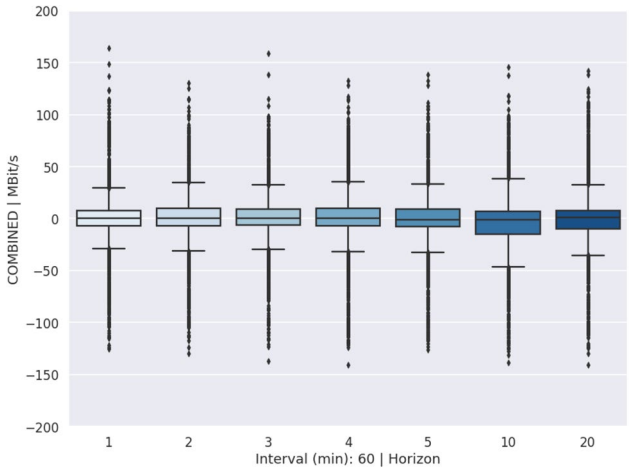

Fig. 10 Error distribution of the COMBINED model

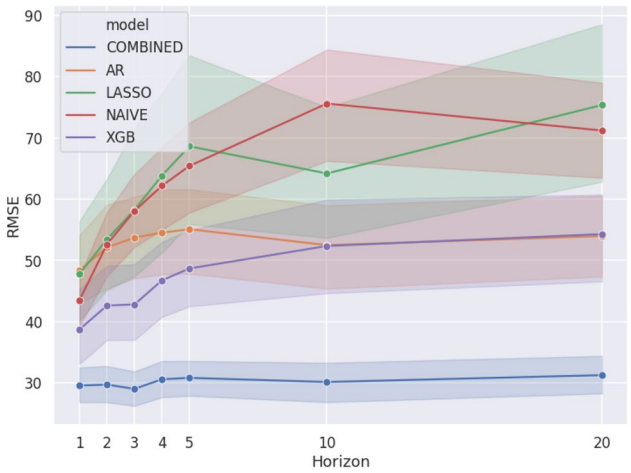

Fig. 11 Distribution of the user-specific RMSE values

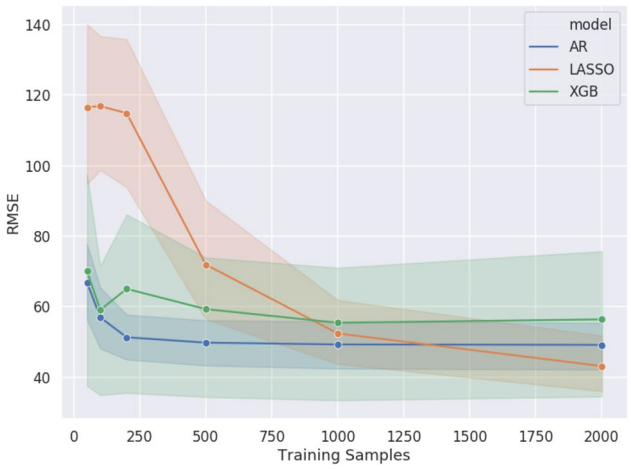

Fig. 12 Distribution of the forecasting error (RMSE) for different training set sizes

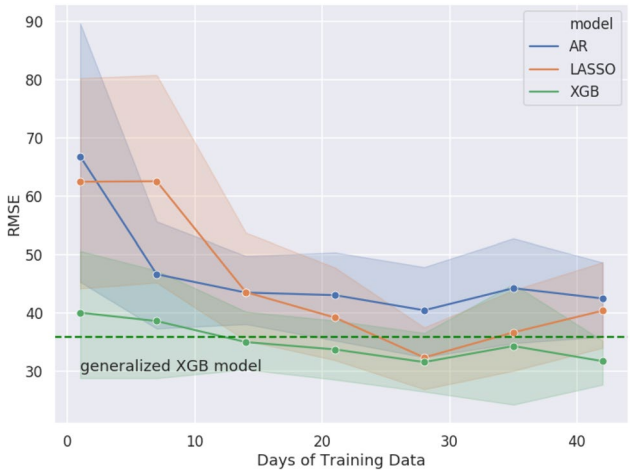

Fig. 13 Distribution of the forecasting errors of a generalized model in comparison to models using the individual users' context data

Acknowledgements Parts of the research in this paper used the LDDC Database made available by Idiap Research Institute, Switzerland and owned by Nokia.

Funding Open Access funding enabled and organized by Projekt DEAL.

References

Abbas N, Zhang Y, Taherkordi A, Skeie T (2018) Mobile edge computing: a survey. IEEE Internet Things J 5(1):450–465

Anagnostopoulos T, Anagnostopoulos C, Hadjiefthymiades S (2011) Mobility prediction based on machine learning. In: Mobile Data Management (MDM), 2011 12th IEEE International Conference on, IEEE, vol 2, pp 27–30

Apple Inc (2019) Siri. https://www.apple.com/ios/ siri/. Accessed 03 Mar 2019

Bonawitz K, Eichner H, Grieskamp W, Huba D, Ingerman A, Ivanov V, Kiddon C, Konecny J, Mazzocchi S, McMahan HB, et al. (2019) Towards federated learning at scale: system design. arXiv preprint arXiv:190201046

Chen T, Guestrin C (2016) Xgboost: A scalable tree boosting system. In: Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, KDD '16, pp 785–794, 10.1145/2939672.2939785, http://doi.acm.org/10.1145/2939672.2939785

Crawdad Project (2019) Crawdad a community resource for archiving wireless data at dartmouth. http://crawdad.org/all-byname.html. Aufgerufen am 03 Mar 2019

Dey AK, Abowd GD (1999) Towards a better understanding of context and context-awareness. Tech. rep. Georgia Institute of Technology, Atlanta

Farris I, Taleb T, Flinck H, Iera A (2018) Providing ultra-short latency to user-centric 5g applications at the mobile network edge. Trans Emerg Telecommun Technol 29(4):e3169

Fernando N, Loke SW, Rahayu W (2013) Mobile cloud computing: a survey. Future Gener Comput Syst 29(1):84–106

Friedman JH (2000) Greedy function approximation: a gradient boosting machine. Ann Stat 29:1189–1232

Hyndman RJ, Athanasopoulos G (2018) Forecasting: principles and practice. OTexts, Heathmont

Laurila JK, Gatica-Perez D, Aad I, Blom J, Bornet O, Do TMT, Dousse O, Eberle J, Miettinen M (2013) From big smartphone data to worldwide research: the mobile data challenge. Pervasive Mobile Comput 9(6):752–771

Lee J (2017) Prediction-based energy saving mechanism in 3g pp nb-iot networks. Sensors 17(9):2008

Lim BY, Dey AK (2010) Toolkit to support intelligibility in context-aware applications. In: Proceedings of the 12th ACM international conference on Ubiquitous computing, ACM, pp 13–22

Mayrhofer R (2004) An architecture for context prediction. PhD thesis, Schriften der Johannes-Kepler-Universität Linz

McMahan HB, Moore E, Ramage D, Hampson S, et al. (2016) Communication-efficient learning of deep networks from decentralized data. arXiv preprint arXiv:160205629

Nicholson AJ, Noble BD (2008) Breadcrumbs: forecasting mobile connectivity. In: Proceedings of the 14th ACM international conference on Mobile computing and networking, ACM, pp 46–57

Nielsen D (2016) Tree boosting with xgboost - why does xgboost win "every" machine learning competition? Dissertation, Norges teknisk-naturvitenskapelige universitet

Nvidia Corporation (2019) Nvidia shield game streaming dienst geforce now. http://https://www.nvidia.com/de-de/shield/games/, http://shield.nvidia.de/game-streaming-with-geforce-now. Aufgerufen am 03 Mar 2019

Orsini G, Bade D, Lamersdorf W (2018a) Cloudaware: empowering context-aware self-adaptation for mobile applications. Trans Emerg Telecommun Technol 29(4):e3210

Orsini G, Bade D, Lamersdorf W (2018b) Generic context adaptation for mobile cloud computing environments (extended version). J Ambient Intell Humaniz Comput 9(1):61–71

Orsini G, Bade D, Lamersdorf W (2016) Generic context adaptation for mobile cloud computing environments. In: The 13th International Conference on Mobile Systems and Pervasive Computing (MobiSPC 2016) / Affiliated Workshops, August 15–18, 2016. Montreal, Quebec, Canada, Elsevier Science, Procedia Computer Science, pp 17–24

Orsini G, Posdorfer W, Lamersdorf W (2019) Efficient mobile clouds: Forecasting the future connectivity of mobile and iot devices to save energy and bandwidth. In: The 14th International Conference on Future Networks and Communications (FNC 2019) / The 16th International Conference on Mobile Systems and Pervasive Computing (MobiSPC 2019) / Affiliated Workshops, August 19-21, 2019, Halifax, Nova Scotia, Canada, Elsevier Science, Procedia Computer Science, vol 155, pp 121–128

Patel P, Ali MI, Sheth A (2017) On using the intelligent edge for iot analytics. IEEE Intell Syst 32(5):64–69

Perera C, Zaslavsky A, Christen P, Georgakopoulos D (2014) Context aware computing for the internet of things: a survey. IEEE Commun Surv Tutor 16(1):414–454

Pokahr A, Braubach L (2013) The active components approach for distributed systems development. Int J Parallel Emerg Distrib Syst 28(4):321–369

Rahman H, Rahmani R (2018) Enabling distributed intelligence assisted future internet of things controller (fitc). Appl Comput Inform 14(1):73–87

Sato K, Fujii T (2017) Radio environment aware computation offloading with multiple mobile edge computing servers. In: 2017 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), IEEE, pp 1–5

Scellato S, Musolesi M, Mascolo C, Latora V, Campbell AT (2011) Nextplace: a spatio-temporal prediction framework for pervasive systems. In: International Conference on Pervasive Computing, Springer, pp 152–169

Shi C, Lakafosis V, Ammar MH, Zegura EW (2012) Serendipity: enabling remote computing among intermittently connected mobile devices. In: Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing, ACM, pp 145–154

Shi C, Pandurangan P, Ni K, Yang J, Ammar M, Naik M, Zegura E (2013) IC-Cloud: Computation offloading to an intermittently-connected cloud. Tech. Rep. GT-CS-13-01, Georgia Institute of Technology

Shiraz M, Gani A (2014) A lightweight active service migration framework for computational offloading in mobile cloud computing. J Supercomput 68(2):978–995

Sigg S (2008) Development of a novel context prediction algorithm and analysis of context prediction schemes. Kassel University Press GmbH, Kassel

Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc Ser B (Methodol) 58(1):267–288