nature portfolio

Peer Review File

Introducing Edge Intelligence to Smart Meters via Federated Split Learning

Corresponding Author: Dr Yi Wang

Parts of this Peer Review File have been redacted as indicated to remove third-party material.

This file contains all reviewer reports in order by version, followed by all author rebuttals in order by version.

Version 0:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
STRENGTHS

+ The paper uses interesting and relevant datasets for the problem of load forecasting.

+ The three-tier end-edge-cloud system design for experimental results is appreciate and an admirable feature of the paper.

+ The authors tackle a timely and relevant area of research.

+ The portion of the work that aims to find an optimal model split using optimization is really interesting.

WEAKNESSES

- The paper is very unfocused and the clarity of the writing suffers because of it.

- Right now, the paper is very bloated. There are a lot of moving pieces for such a short work that the important pieces and takeaways feel lost.

- Some plots are missing information to understand them.

- There are concerns about the validity and significance of the results—particularly with regard to the communication cost reduction.

- In numerous occasions, the authors state this is the first work to consider edge intelligence (namely federated learning and split learning) for smart meters. This is not true. This problem has been explored by other works (see detailed comments for references).

DETAILED COMMENTS

- In terms of writing, I would recommend a major revision regarding the paper's writing with a more coherent follow-through with the overall objective. In general, the paper is throwing together several big ideas into one project that is summarized very briefly. From federated learning, to advanced metering infrastructure (i.e., smart meters), to split learning, to knowledge distillation, asynchronous aggregation, and optimal model splitting... it is difficult to truly grasp what is the central problem. As such, the paper feels very scattered and is very difficult to follow. I struggled to make sense of all the moving pieces and the subject matter of the paper is something I am already very familiar with. Some suggestions are provided below:
- The Introduction section reads very much like a related work section rather than an introduction. While it's important to highlight relevant works (though, this feels neglected due to many missing works that are very relevant), the authors spend a large bulk of their introduction touching on other problems in federated learning that are not relevant to their work (e.g., model/gradient compression) or jumping from challenge to challenge in a very disconnected manner. For instance, there is

no transition from model compression to knowledge distillation despite them being adjacent sentences.
- Many concepts are mentioned and not adequately defined. For instance, knowledge distillation is mentioned as a communication reduction technique without a brief summary as to what the goal of knowledge distillation is. Even a brief sentence would suffice (e.g., "Knowledge distillation is a problem in deep learning of trying to transferring or 'distilling' knowledge from one model to another—usually a larger model to a smaller model").
- There are sentences that need complete revisions. For instance, sentence 3 on page 3 reads very clumsily, "Future, in order to make full use of smart meters..."
- Minor grammatical mistakes with misused articles (e.g., unnecessary uses of "the") and capitalization ("We" is incorrectly capitalized in Section 2.4).
- Key concepts are not well-introduced or explained. Most notably, the authors quickly mention smart meters at the start of the paper. They explain that they are ubiquitous but never clearly summarize what they are or what they do.
- The acronym AMI (advanced metering infrastructure) is introduced, but only used once. This is also done for SRAM (static random access memory) in the abstract—though its used in the body of the paper.

- The authors claim in the abstract that they are able to achieve "superior forecasting accuracy compared to resource-unlimited methods" in the abstract of the paper. This is a *very* bold claim and the paper does not really demonstrate this. Specifically, the analysis of their learning tasks in the paper are very limited because the authors consider extremely small neural networks with 3-5 hidden layers. These models are so small that "resource-unlimited" systems are not even necessary to consider, let alone claim superiority to.

- The results, at the moment, are not very compelling. Specifically in Supplementary Figure 1, the predictions provided by all of the approaches seem to perform near identically. What is the real impact of such a relatively small discrepancy in predictive performance for a regression task? Will end users of a home with a smart meter in an accuracy difference of 0.14 (Fig. 4, top left, 3 layers vs. 5 layers)?

- Continuing the discussion on results, the result that looks most significant is the reduction is communication cost. However, this result feels very unintuitive when thinking it over. In Fig. 5, there appears to be a massive reduction in communication cost when you compare synchronous to semi-asynchronous and asynchronous. From my understanding of the experimental setup, the number of rounds for each experiment is fixed at 100 rounds and the experiment runs until all 100 rounds complete. First, this choice feels a bit odd and in favor of asynchronous execution in general. A better stopping condition would be converged testing loss against the fixed test data set. However, the other issue is that the communication cost reduction doesn't make sense. In the semi-asynchronous case, the end-edge aggregation is synchronous and the edge-cloud aggregation is asynchronous. If every end-edge subtree runs for 100 rounds, then wouldn't the total training time simply be the training time of the slowest cluster of smart meters multiplied by 100? If so, then there should be no noticeable reduction in either case because the slower cluster of end devices is going to be the bottleneck that ultimately determines the total training time regardless of the other end-edge clusters continuing on asynchronously. If I'm misunderstanding, then this should this be better clarified in writing.

- Regarding the method for finding the optimal model split, it would be interesting to see this solution run on larger models—even if it means ignoring the memory constraints of the MCU nodes. This is generally just an interesting research problem in split learning. But, the fact the authors only focus on very small models with 3-5 hidden layers, it becomes exceedingly difficult to be convinced that the solution is able to make a good decision. For instance, I imagine that in the case of 3 hidden layers, a naïve random solution has a 33% chance of making the optimal model split.

- The decision to cluster the smart meters by compute power using a (balanced) K-means algorithm feels very unintuitive. In hierarchical FL settings, the hierarchy generally is established based on the true underlying network. For instance, in a paper by Hudson et al. on FL for the nonintrusive load monitoring problem, they consider 3-tier federated aggregation of smart metering data where the middle aggregator (analogous to the edge cloud in the authors' submitted manuscript) is localized to a neighborhood in the neighborhood area network in the AMI system and it is connected to home area networks (each with a smart meter) most local to it. This seems more natural for hierarchical networks. Additionally, as the authors have mentioned, smart meters are notoriously low-power in terms of compute availability. Clustering them based on compute capacity seems less important since they likely do not vary that widely anyway. Clustering them based on geographic distance is more intuitive from a networks perspective. Finally, it is more logical to apply this approach due to the naturally non-iid data distributions of energy consumption across different neighborhoods. For instance, more affluent neighborhoods are likely going to have homes with different energy consumption patterns than homes in a poor neighborhood.

- In Table 2, FedAvg-M is the next best algorithm based on the authors' metrics. I would be curious to see how the FedProx algorithm performs against the authors' proposed solution. This federated aggregation algorithm adds a proximal term (based on the norm between the global model and the locally-trained model) to the loss before backprop. This "grounds" the locally-trained models to not stray too far apart from the global model and has shown well to work on non-iid data. It might make for a more apt comparison to the knowledge distillation-driven approach proposed by the authors.

- Plots need to be clearer. More specifically, plots often do not have clear titles or subtitles to clarify what they are communicating. One such example is Figure 3(b). The individual subplots in this figure are not clearly labeled so it is unclear what distinguishes these 4 subplots from one another. Another comment on clear plotting can be said about the choice to use "split ratio" in Figure 3(a). This is a very inaccessible metric (*especially* without the supplemental text). It might be clearer to just mention the layer(s) that split and placed on the edge server—this seems to be indicated by the color, but the ratio is a confusing detail.

- The authors state that this is the first work to consider edge intelligence for smart meters. This is not true and other relevant works should be adequately highlighted—with the authors providing a clear distinction between their work and what's already been done. Below are some examples of papers exploring this topic:
- Fekri, Mohammad Navid, Katarina Grolinger, and Syed Mir. "Distributed load forecasting using smart meter data: Federated learning with Recurrent Neural Networks." _International Journal of Electrical Power & Energy Systems_137 (2022): 107669.
- Fekri, Mohammad Navid, Katarina Grolinger, and Syed Mir. "Asynchronous adaptive federated learning for distributed load forecasting with smart meter data." _International Journal of Electrical Power & Energy Systems_ 153 (2023): 109285.
- Hudson, Nathaniel, Md Jakir Hossain, Minoo Hosseinzadeh, et al. "A framework for edge intelligent smart distribution grids via federated learning." _2021 International Conference on Computer Communications and Networks (ICCCN)_. IEEE, 2021.
- Wang, Yi, Ning Gao, and Gabriela Hug. "Personalized federated learning for individual consumer load forecasting." _CSEE Journal of Power and Energy Systems_ 9.1 (2022): 326-330.
- Taïk, Afaf, and Soumaya Cherkaoui. "Electrical load forecasting using edge computing and federated learning." _ICC 2020-2020 IEEE international conference on communications (ICC)_. IEEE, 2020.

Reviewer #2

(Remarks to the Author)
In this paper, the authors investigated the potential of smart meters for supporting the excavation of demand-side flexibility by using edge intelligence. Generally, the data received by a single smart meter is limited while sharing data directly in the network causes the leakage of the data, introducing privacy issues. From this perspective, the authors leveraged federated learning. In this way, only the model parameters would be shared, instead of the original data, improving data privacy. Considering the resource limitations due to the physical size of smart meters, the authors decided to seek help from the edge server to perform local training via split learning. In this way, the smart meter only executed partial model layers and delegated the heavy training processes to the nearby edge server. Combining split learning and federated learning, the authors proposed the end-edge-cloud federated split learning framework to achieve smart grids. In addition, the authors implemented a hardware platform to evaluate their approach with other representative approaches.

As a researcher in edge intelligence, I am really happy to see the implementation of edge intelligence in the industry and also appreciate the effort of the authors in implementing the hardware platform. Such study is needed but there are various places that require clarification and further consideration to make the article more convincing and comprehensive.

While split learning (SL) is in general efficient in reducing overall training time by delegating heavy training computation on powerful edge servers (compared to smart meter), the training time on edge servers is not further analysed. Offloading too many training tasks to edge servers may introduce an extra bottleneck in the system. To eliminate such risk, the authors are suggested to add the overhead analysis to figure this out and explicitly demonstrate it to the readers. If it becomes an issue, I suggest using pipeline-based approaches to schedule learning tasks on smart meters and edge servers in a more flexible way.

Another concern is still about SL. The authors proposed a ratio to determine the way to split the model to minimize the overall training time while fulfilling the memory constraints. However, the layers selected based on the ratio are not explicitly discussed. In many machine learning models, the characteristics of different layers are different. The layers closer to the input are more important to the feature extraction and the layers closer to the output are more important to the feature fusion and integration. For example, in the case that three of the seven layers are put on the smart meter, what's the performance of (1, 2, 7), (1,4,7), (1,6,7) or others? More evidence could be provided to enhance the feasibility of this approach in real-world scenarios.
As mentioned by the authors (also well acknowledged in the community), edge devices (including smart meters) are also resource-constrained. In this work, the authors used the meter with only 192KB of SRAM. To analyse the memory limitation of the smart meter, the authors mentioned several types of memory usage in Section 4.1. However, the storage usage of training data is not mentioned. How much memory will those data occupy? If the data needs more storage resources, will the model training be impacted? Since this is an online training framework, more data will be collected. Will this make the memory issue more serious?

The last point is about privacy protection via federated learning. However, this is more related to computer science rather than engineering. In edge intelligence, there exists a number of studies that successfully infer partial data based on model parameters. Thus, the paper can be more concise to say enhancing privacy (the authors did this in several places) but not be too confident in data protection.

There are also some written issues to be fixed such as missing descriptions of notations and abbreviations.

In conclusion, I recommend a major revision.

Reviewer #3

(Remarks to the Author)
The paper provides an end-edge-cloud federated split learning framework model to enable training on resource-constrained smart meters. Imho, the work is more like an algorithm improvement instead of a widely impacted study. It may be a good

technical paper, but the work done is not in the style of Natural Communication research.

Strength
1) The paper gives a detailed consideration of the intellectualization of smart meters.
2) It is commendable that the work provides hardware platform validation of the proposed method.

Weakness
1) The concepts, e.g. federated learning, and edge intelligence, are not new. Distributed learning/training structure and privacy concerns have been extensively studied in smart grid and communication fields.
2) Will on-device intelligence for smart meters bring a lot of energy consumption?
3) Only CNN and LSTM were tested as the benchmark, which weakens the significance of this work in terms of deep learning efficiency.
4) The two test datasets are all before 2018, which is not in a kind of up-to-date way.
5) Although the hardware platform was provided, it's more like a demo with a very limited scale. It would be more meaningful if the technique could step out of the lab.
6) There is not a strong connection between the smart meter and renewable energy. It's better to make it clear at what extent/how large the improvement is of smart meter-supported demand flexibility for the RES promotion.

Version 1:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
The authors' implementation of a hardware testbed for edge intelligent smart meters is notable and is a good contribution to the field. I feel satisfied with the authors' response to my original comments. Their alterations are comprehensive and significant.

Reviewer #2

(Remarks to the Author)
I appreciate the efforts spent by the authors on revising this paper and addressing my questions.

Compared to the previous version, the quality of this manuscript is improved. However, there are still several points to be considered:

The limitations of the communication and computational capabilities of edge devices are well-known challenges in the edge environment. Thus, there are a number of existing studies focusing on communication and computational resource optimization, especially papers published in IEEE Transactions, and many international conferences. Thus, the statement "previous studies have often overlooked the limitations of the communication and computational capabilities of edge devices" in the last paragraph on page 3 is not proper. I agree that the implementation of smart meters would be harder than other edge devices, however, the authors need to explicitly demonstrate the special challenge their approach tackled.

In addition, the contents in Fig. 1 are not exactly the same as the statements in the paper. The authors declare the privacy advantage achieved by federated learning, but this is not mentioned in the overview. In addition, the overlapping part is supposed to be the common characteristics or advantages of federated learning and split learning. However, this figure should clearly show how federated learning and split learning can enhance the performance of each other in various dimensions. The current version needs to be modified.

Another point is about privacy-enhancing performance. The metrics about privacy are not clear in the experiments. Generally, privacy performance can be measured in multiple ways, such as whether the desired data can be figured out by privacy attacks like membership inference attacks, etc. As the authors claim that this is part of their contributions, it would be necessary to explicitly demonstrate the results to the readers.

The last point is about the limitations of their approach. The authors provide their discussions in Section 3, which is good. However, the proposed approach has clearly limitations in the implementations and also the techniques adopted. I understand that it would be hard to have a perfect design, however, the discussions about the approach limitations are also a significant contribution and key idea to be delivered to the readers. Thus, I would recommend the authors provide such discussions to enhance the quality of this paper.

Reviewer #3

(Remarks to the Author)
The authors provide good responses to my concerns. I have some further but relatively minor comments in this turn.
1) Even though I could see there is a relatively light energy requirement for the smart meters, the comparison of power consumption between the smart meter and household energy is kind of not at the same level. Smart meters + cloud server +

edge serve VS total house energy saving will be a better comparison. In this case, is there a possibility that the consumed energy by the whole system (cloud + edge servers + meter) is larger than the saved energy? I believe there is a need for the scale balance here.
2) IMHO, it's more convincing to compare the energy consumption between the conventional way (e.g. centralized computation/learning method) and edge intelligence FL-based smart meter method if you wanna show the reduced energy consumption. Do you have any technical support for the assumption of '1/24 maximum power operation of the smart meters'?
3) How did you calculate the reduced electricity cost, increased renewable energy accommodation, and reduced carbon emission?
4)'To quantify the impact of edge intelligence on downstream …' I got confused here. From my understanding, successful energy management is based on the data/energy profile collected by the smart meters. Edge intelligence could be a way to realize energy management but not the only way. What is the role of the federated splitting learning-based edge intelligence here?

Version 2:

Reviewer comments:

Reviewer #2

(Remarks to the Author)
The authors have addressed my previous comments and I really enjoyed going through the revised paper.

Reviewer #3

(Remarks to the Author)
The authors have properly responded my comments. I don't have further comments in this turn.

Reviewer 1 (Remarks to the Author):

STRENGTHS

+ The paper uses interesting and relevant datasets for the problem of load forecasting.

+ The three-tier end-edge-cloud system design for experimental results is appreciate and an admirable feature of the paper.

+ The authors tackle a timely and relevant area of research.

+ The portion of the work that aims to find an optimal model split using optimization is really interesting.

Reply:

Thank you for the positive feedback on our effort to demonstrate the superiority of the proposed method. We have looked into your comments carefully and revised the manuscript accordingly. We will respond point-by-point to your comments and questions in the following.

WEAKNESSES

- The paper is very unfocused and the clarity of the writing suffers because of it?

Reply:

Thanks for your comment. In order to enhance the clarity of the paper, we have provided an overview scheme to illustrate the main problem to be solved. The demonstration of key concepts and relevant research studies are added to better emphasize the core of this paper. Modifications can be found in the response to your detailed comment 1.

- Right now, the paper is very bloated. There are a lot of moving pieces for such a short work that the important pieces and takeaways feel lost.

Reply:

Thanks for your suggestion. We aim to bring edge intelligence from theory to real-world practice, which requires a comprehensive consideration of hardware constraints (memory, computation power, and communication capacity) and data privacy issues that result in the need for several techniques. These techniques are logically integrated into the proposed framework to solve the main problem in this paper. For better clarity, we have provided an overview scheme to demonstrate the relations between the problems and the proposed solutions. Modifications can be found in the response to your detailed comments 1, 2, and 5. Furthermore, a paragraph of takeaways is added to highlight the main discoveries from this study as follows:

This study provides the following takeaway messages. First, implementing edge intelligence algorithms on smart meters should primarily consider hardware resource availability for the feasibility of grid applications. Second, smart meters can collaborate through federated learning to improve energy analytics performance in edge intelligence by orchestrating the cooperation of distributed data resources. Third, smart meters can split large-scale models with the assistance of high-capacity servers to improve energy analytics performance in edge intelligent systems by orchestrating the cooperation of hierarchical computational resources. Finally, edge intelligence on smart meters can substantially optimize energy management, promote sustainable energy development, and thereby advance the decarbonization of power and energy systems.

- Some plots are missing information to understand them.

Reply:

Thank you for pointing out that some plots are missing information that is necessary for understanding. We apologize for any confusion caused by the lack of information in some plots. We have revised the figures and added the necessary information to ensure clarity and comprehensibility. Details can be found in the responses to your detailed comments 9, 11, and 13.

- There are concerns about the validity and significance of the results—particularly with regard to the communication cost reduction.

Reply:

Thank you for your valuable comments. We have added more experiments in the areas of benchmark methods, model depth, consumer scale, model backbone, and economic benefits to demonstrate the effectiveness of our proposed method. Specifically, we have added Supplementary Fig. 7 and Supplementary Fig. 8 to provide more experimental details on asynchronous aggregation, which helps us better understand our method's advantages in reducing communication overhead. Modifications can be found in the responses to your detailed comments 7 to 12.

- In numerous occasions, the authors state this is the first work to consider edge intelligence (namely federated learning and split learning) for smart meters. This is not true. This problem has been explored by other works (see detailed comments for references).

Reply:

We sincerely respect the reviewer's opinion, but we would like to claim that existing work on edge intelligence is mainly utilizing the smart meter data to carry out simulation experiments instead of truly implementing their methods on smart meter hardware. The main point that

differentiates our work from the existing work is that we have solved the resource-constrained problem of smart meters and showcased the effectiveness of the proposed method on a hardware platform. To the best of our knowledge, this represents the first attempt to transition the concept of intelligence on smart meters from theory to tangible practice.

We have enriched the literature review to highlight relevant work that uses smart meter data for edge intelligence and provided a clearer distinction between our work and existing work. Modifications are shown in the response to the detailed comment 14.

DETAILED COMMENTS

1. In terms of writing, I would recommend a major revision regarding the paper's writing with a more coherent follow-through with the overall objective. In general, the paper is throwing together several big ideas into one project that is summarized very briefly. From federated learning, to advanced metering infrastructure (i.e., smart meters), to split learning, to knowledge distillation, asynchronous aggregation, and optimal model splitting... it is difficult to truly grasp what is the central problem. As such, the paper feels very scattered and is very difficult to follow. I struggled to make sense of all the moving pieces and the subject matter of the paper is something I am already very familiar with. Some suggestions are provided below:

Reply:

We gratefully thank the reviewer for the insightful comments. The central problem considered in this paper is **how to effectively utilize distributed data resources to train complex and accurate load forecasting models on resource-constrained smart meters to realize edge intelligence in smart grids**. We tackle such a central problem by answering two sub-questions: first is 'How can we efficiently utilize distributed data?' and second is 'How can we train models on resource-constrained devices?'. For better clarity, we present an overview of the problems to be solved and corresponding proposed solutions in Fig. 1. We investigated federated learning for the first question and proposed hardware-based clustering and semi-asynchronous aggregation methods. We looked into split learning for the second problem and proposed optimal splitting and collaborative knowledge distillation strategies. We integrated these techniques into an end-edge-cloud federated split learning framework, which results in higher accuracy, reduced memory footprint, faster computation speed, and smaller communication overhead.

We have added these contents to the revised manuscript and hope this can help the reviewer and readers understand the logic of our work better.

While previous studies have often overlooked the limitations of the communication and computational capabilities of edge devices, our work addresses the challenge of translating these methods into practical, real-world applications tailored for smart meter hardware. We focus

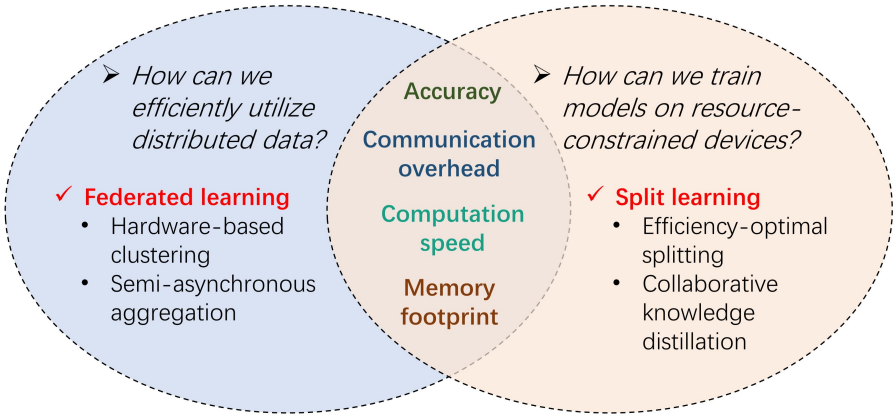


Fig. 1: Overview of the problems to be solved and the corresponding proposed solutions. This paper focuses on answering two critical questions in achieving on-device intelligence: 'How can we efficiently utilize distributed data?' and 'How can we train models on resource-constrained devices?'. We investigate federated learning to address the first question and propose hardware-based clustering and semi-asynchronous aggregation methods. We consider split learning to address the second question and propose optimal splitting and collaborative knowledge distillation strategies.

on overcoming the constraints inherent to edge environments, ensuring that our solutions are not only theoretically sound but also viable for on-the-ground deployment. We present an overview of the problems to be solved and the corresponding proposed solutions in Fig. 1. This paper focuses on answering two critical questions in achieving on-device intelligence: 'How can we efficiently utilize distributed data?' and 'How can we train models on resource-constrained devices?'. To answer the above questions, we present an end-edge-cloud framework that combines federated learning and split learning to intellectualize resource-constrained smart meters for on-device load forecasting in a privacy-enhancing manner. This framework consolidates several properties: higher accuracy, reduced memory footprint, faster computation speed, and smaller communication overhead.

---

2. The Introduction section reads very much like a related work section rather than an introduction. While it's important to highlight relevant works (though, this feels neglected due to many missing works that are very relevant), the authors spend a large bulk of their introduction touching on other problems in federated learning that are not relevant to their work (e.g., model/gradient compression) or jumping from challenge to challenge in a very disconnected manner. For instance, there is no transition from model compression to knowledge distillation despite them being adjacent sentences.

Reply:

Thank you for the comments on the introduction section. We have organized the introduction

section by first discussing the pivotal role of smart meters in supporting demand-side flexibility to account for the decarbonization issue. Then, we point out the limitations of existing smart meters and the necessity to achieve edge intelligence. Subsequently, we have reviewed the existing work to tackle two main challenges in achieving edge intelligence. The first challenge is how to carry out complex computations on resource-constrained smart meters, where the concept of split learning is introduced. The second challenge is how to effectively utilize distributed data resources, where federated learning is introduced. Then, we pointed out the challenges of implementing FL in large-scale deployment of smart meters, where communication and model aggregation efficiency are the key issues. Lastly, we demonstrate the difference between our study with existing work and conclude the problems solved in this paper by providing an overview scheme. The main experimental results are also presented to demonstrate the effectiveness of the proposed methods. To this end, we believe the introduction section provides a comprehensive discussion about **'why'** and **'how'** to achieve edge intelligence on smart meters, instead of a simple listing of related work. We have reorganized the introduction section to show a clearer logic.

Besides, we sincerely thank the reviewer's suggestion for reviewing the related work. We have replaced the weak-relevant references (model/gradient compression) with more relevant references on federated learning for edge intelligence. The modifications are shown as follows.

Several studies have investigated FL for edge intelligence, such as [1-5]. However, these studies mainly utilize smart meter data to carry out simulation experiments instead of implementing their methods on resource-constrained smart meter hardware. There is still a lack of a unified framework that considers all perspectives of model accuracy, on-device memory footprint, computation speed, and communication overhead to fully achieve on-device intelligence.

While previous studies have often overlooked the limitations of the communication and computational capabilities of edge devices, our work addresses the challenge of translating these methods into practical, real-world applications tailored for smart meter hardware. We focus on overcoming the constraints inherent to edge environments, ensuring that our solutions are not only theoretically sound but also viable for on-the-ground deployment. We present an overview of the problems to be solved and the corresponding proposed solutions in Fig. 1. This paper focuses on answering two critical questions in achieving on-device intelligence: 'How can we efficiently utilize distributed data?' and 'How can we train models on resource-constrained devices?'. To answer the above questions, we present an end-edge-cloud framework that combines federated learning and split learning to intellectualize resource-constrained smart meters for on-device load forecasting in a privacy-enhancing manner. This framework consolidates several properties: higher accuracy, reduced memory footprint, faster computation speed, and smaller communication overhead.

[1] Fekri, Mohammad Navid, Katarina Grolinger, and Syed Mir. ”Distributed load forecasting using smart meter data: Federated learning with Recurrent Neural Networks.” International Journal of Electrical Power& Energy Systems, 137 (2022): 107669.

[2] Fekri, Mohammad Navid, Katarina Grolinger, and Syed Mir. ”Asynchronous adaptive federated learning for distributed load forecasting with smart meter data.” International Journal of Electrical Power& Energy Systems, 153 (2023): 109285.

[3] Hudson, Nathaniel, Md Jakir Hossain, Minoo Hosseinzadeh, et al. ”A framework for edge intelligent smart distribution grids via federated learning.” 2021 International Conference on Computer Communications and Networks (ICCCN), IEEE, 2021.

[4] Wang, Yi, Ning Gao, and Gabriela Hug. ”Personalized federated learning for individual consumer load forecasting.” CSEE Journal of Power and Energy Systems, 9.1 (2022): 326-330.

[5] Taïk, Afaf, and Soumaya Cherkaoui. ”Electrical load forecasting using edge computing and federated learning.” ICC 2020-2020 IEEE international conference on communications (ICC), IEEE, 2020.

---

3. There are sentences that need complete revisions. For instance, sentence 3 on page 3 reads very clumsily, ”Future, in order to make full use of smart meters...”

Reply:

We thank the reviewer for the careful review. We have completely revised sentences in the manuscript with the help of an editing service provided by Springer Nature (Certification can be found in the Supportive Figure).

---

4. Minor grammatical mistakes with misused articles (e.g., unnecessary uses of ”the”) and capitalization (”We” is incorrectly capitalized in Section 2.4).

Reply:

We thank the reviewer for the careful review. We have corrected the typo and completely revised sentences in the manuscript with the help of the editing service provided by Springer Nature (Certification can be found in the Supportive Figure).

---

5. Key concepts are not well-introduced or explained. Most notably, the authors quickly mention smart meters at the start of the paper. They explain that they are ubiquitous but never clearly summarize what they are or what they do.

Reply:

Thank you for your valuable comments. Smart meters are advanced energy meters that have

[REDACTED]

Supportive Figure: Springer Nature editing certification.

replaced traditional low-resolution mechanical meters and become the core part of the advanced metering infrastructure in power systems. These advanced meters are supported by several sensors, control devices, and dedicated communication infrastructure. Smart meters can record real-time energy information from the demand side, including voltage, frequency, and energy consumption, and enable bidirectional communication between system operators and end-users. Smart meters play a pivotal role in promoting renewable energy accommodation by providing data and hardware foundations to harness demand-side flexibility. On one hand, smart meter data enables the estimation of demand response potential [4] and dynamic pricing design [5] to integrate renewable energy. On the other hand, smart meters can act as agents for home energy management systems to monitor the distributed renewable energy generation, storage and consumption [6]. We have added this information to the revised manuscript.

Harnessing demand-side flexibility is a cost-effective strategy for promoting renewable energy accommodation [1], where smart meters play a pivotal role in this process. Smart meters are the core part of the advanced metering infrastructure in power systems, which are supported by sensors, control devices, and dedicated communication infrastructure [2]. Smart meters can record real-time energy information, including voltage, frequency, and energy consumption, from the demand side and can enable bidirectional communication between system operators

and end-users [3]. The advanced functions of smart meters provide a strong foundation for harnessing demand-side flexibility in terms of data and hardware platforms [4,5]. On the one hand, smart meter data enable the estimation of demand response potential [6] and dynamic pricing design [7] to integrate renewable energy. On the other hand, smart meters can act as agents for home energy management systems to monitor the distributed renewable energy generation, storage, and consumption [8].

[1] O'Shaughnessy E, Shah M, Parra D, et al. The demand-side resource opportunity for deep grid decarbonization[J]. Joule, 2022, 6(5): 972-983.

[2] Avancini D B, Rodrigues J J P C, Martins S G B, et al. Energy meters evolution in smart grids: A review[J]. Journal of cleaner production, 2019, 217: 702-715.

[3] Mohassel R R, Fung A, Mohammadi F, et al. A survey on advanced metering infrastructure[J]. International Journal of Electrical Power & Energy Systems, 2014, 63: 473-484.

[4] Barai G R, Krishnan S, Venkatesh B. Smart metering and functionalities of smart meters in smart grid-a review[C] 2015 IEEE Electrical Power and Energy Conference (EPEC). IEEE, 2015: 138-145.

[5] Wang Y, Chen Q, Hong T, et al. Review of smart meter data analytics: Applications, methodologies, and challenges[J]. IEEE Transactions on Smart Grid, 2018, 10(3): 3125-3148.

[6] Dyson M E H, Borgeson S D, Tabone M D, et al. Using smart meter data to estimate demand response potential, with application to solar energy integration[J]. Energy Policy, 2014, 73: 607-619.

[7] Cai Q, Xu Q, Qing J, et al. Promoting wind and photovoltaics renewable energy integration through demand response: Dynamic pricing mechanism design and economic analysis for smart residential communities[J]. Energy, 2022, 261: 125293.

[8] Zhou B, Li W, Chan K W, et al. Smart home energy management systems: Concept, configurations, and scheduling strategies[J]. Renewable and Sustainable Energy Reviews, 2016, 61: 30-40.

---

6. The acronym AMI (advanced metering infrastructure) is introduced, but only used once. This is also done for SRAM (static random access memory) in the abstract—though its used in the body of the paper.

Reply:

Thank you for the careful review. We have deleted the unnecessary acronym AMI in the manuscript. We keep the acronym SRAM in the manuscript since it is referred to twice in the body of the

paper.

---

*7. The authors claim in the abstract that they are able to achieve "superior forecasting accuracy compared to resource-unlimited methods" in the abstract of the paper. This is a *very* bold claim and the paper does not really demonstrate this. Specifically, the analysis of their learning tasks in the paper are very limited because the authors consider extremely small neural networks with 3-5 hidden layers. These models are so small that "resource-unlimited" systems are not even necessary to consider, let alone claim superiority to.*

Reply:

Thank you for commenting on our statement, which indeed required clarification. We have changed the statement of 'resource-unlimited methods' to 'conventional methods trained on high-capacity servers' in the manuscript. Here, we aim to compare the performance of the proposed method with some common memory-intensive methods (such as FedAvg-M) that can only be trained on the server. Furthermore, the reason we only considered 5-layer neural networks is that load prediction tasks generally do not require intricate, deep networks widely used in image recognition tasks. As suggested by the reviewers, these methods, including ours, are expected to utilize deeper neural networks to improve forecasting accuracy. Hence, we conducted extensive experiments on all the methods considering a complete network with 7 hidden layers, where the results are reported in Supplementary Table 3.

Supplementary Table 3: Performance of different methods on BDG2 and CBTs in terms of accuracy, memory, training time, and communication overhead per round considering deeper neural network with 7 hidden layers.

Method	BDG2			CBTs			Memory (KB)	Training Time (s)	Communication (KB)
	RMSE	MAPE	MAE	RMSE	MAPE	MAE			
Local-M	22.75	7.68	8.03	0.4678	26.68	0.2721	5178.25 (1.0x)	4335.86 (1.01x)	-
FedAvg-M	22.33	**6.84**	7.44	**0.4617**	26.28	0.2638	5178.25 (1.0x)	4387.33 (1.0x)	761.5 (1.0x)
FedProx-M	22.42	7.01	7.52	0.4620	26.17	0.2641	5178.25 (1.0x)	4387.33 (1.0x)	761.5 (1.0x)
Split	22.93	7.89	8.14	0.4694	26.74	0.2713	103.75 (49.9x)	281.86 (15.55x)	91.25 (8.34x)
SFLV1	22.56	7.03	7.57	0.4963	26.43	0.2650	103.75 (49.9x)	282.84 (15.51x)	96.75 (7.87x)
SFLV2	22.75	7.18	7.66	0.4647	26.39	0.2663	103.75 (49.9x)	282.84 (15.51x)	96.75 (7.87x)
Proposed	**22.22**	6.86	**7.38**	0.4626	**26.12**	**0.2637**	115.07 (45x)	214.68 (20.43x)	74.43 (10.23x)

Comparing the results in Tables 1 and Supplementary Table 3, we can observe that employing deeper networks results in a rather limited or even negative gain in accuracy. In addition, for non-split methods like Local-M, FedAvg-M, and FedProx-M, deeper networks imply a greater memory footprint, training time, and communication overhead for smart meters. Owing to the assistance of edge servers in the proposed framework, the burden on the smart meter does

Table 1: Performance of different methods on BDG2 and CBTs in terms of accuracy, memory, training time, and communication overhead per round.

Method	BDG2			CBTs			Memory (KB)	Training Time (s)	Communication (KB)
	RMSE	MAPE	MAE	RMSE	MAPE	MAE			
Cen	22.82	8.41	8.20	0.4780	26.98	0.2727	2578.42 (1.0×)	586.42 (2.05×)	-
Local-S	23.56	8.13	8.12	0.4698	28.19	0.2726	152.53 (16.9×)	41.19 (29.14×)	-
FedAvg-S	22.79	7.67	7.98	0.4681	27.04	0.2699	152.53 (16.9×)	46.49 (28.80×)	5.50 (27.18×)
FedProx-S	22.58	7.71	7.91	0.4668	27.17	0.2678	152.53 (16.9×)	46.49 (28.80×)	5.50 (27.18×)
Local-M	22.84	7.75	8.03	0.4645	26.67	0.2682	2578.42 (1.0×)	1187.15 (1.01×)	-
FedAvg-M	22.25	**6.96**	7.48	**0.4614**	25.81	0.2637	2578.42 (1.0×)	1200.44 (1.0×)	149.50 (1.0×)
FedProx-M	22.21	7.16	7.62	0.4622	25.76	0.2639	2578.42 (1.0×)	1200.44 (1.0×)	149.50 (1.0×)
Split	23.27	7.68	7.96	0.4679	26.83	0.2687	103.75 (24.8×)	96.00 (12.50×)	91.25 (1.63×)
SFLV1	22.34	7.25	7.68	0.4647	26.03	0.2664	103.75 (24.8×)	96.49 (12.44×)	96.75 (1.54×)
SFLV2	22.76	7.53	7.89	0.4674	26.49	0.2683	103.75 (24.8×)	96.49 (12.44×)	96.75 (1.54×)
Proposed	**22.17**	6.98	**7.44**	0.4630	**25.74**	**0.2636**	115.07 (22.4×)	62.41 (19.23×)	74.43 (2.01×)

not increase with the depth of neural networks, as there are still only a few layers deployed on the smart meter, and the additional model layers are taken up by the edge servers. Remarkably, the results show that the proposed method also achieves the best performance among existing load forecasting methods when considering a deeper neural network with 7 hidden layers. Our framework effectively reduces peak memory by 45x, training time by 20.43x, and communication overhead by 10.23x.

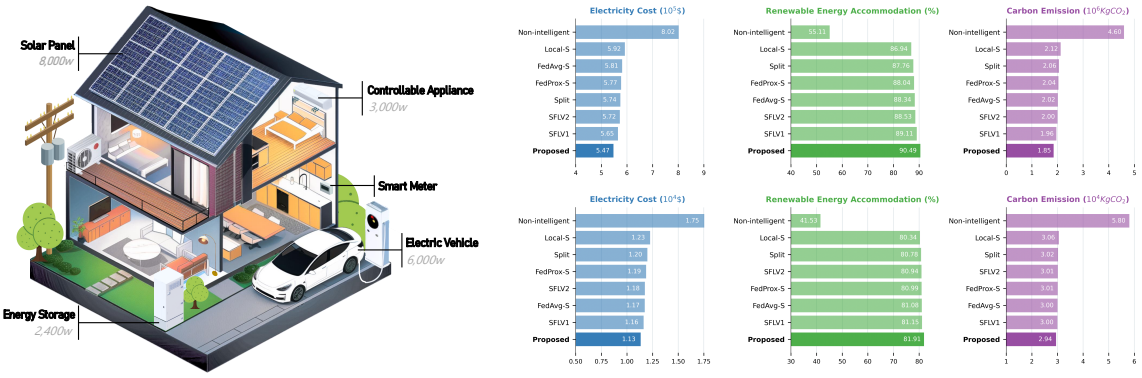
---

8. The results, at the moment, are not very compelling. Specifically in Supplementary Figure 1, the predictions provided by all of the approaches seem to perform near identically. What is the real impact of such a relatively small discrepancy in predictive performance for a regression task? Will end users of a home with a smart meter in an accuracy difference of 0.14 (Fig. 4, top left, 3 layers vs. 5 layers)?

Reply:

Thanks for your attention to the impact of forecasting results on downstream decision-making processes. First of all, we would like to emphasize that the superiority of the proposed method is manifested in the overall improvement of both accuracy and efficiency. Note that the enhancement in the efficiency metrics can effectively ensure the feasibility of the proposed method for smart meter intelligence. To be specific, the peak memory footprint, as a hard constraint, determines whether the model can be trained without overflowing the SRAM in smart meters. Furthermore, the reduction in training time and communication overhead can improve the response speed of the smart grid and alleviate network congestion and delay.

Regarding the improvement in accuracy, although the prediction curves of different methods in Supplementary Fig. 1 are quite close, the slight increase in prediction accuracy brings about quite significant economic benefits for power grid scheduling and management. To quantify the impact of edge intelligence on downstream decision-making, we further investigated the impact of the proposed on-device forecasting approach on individual household energy management and demonstrated its great effectiveness in **reducing electricity cost (31.79%), promoting renewable energy accommodation (35.38%), and lowering carbon emissions (59.78%).** The following discussions for experimental results have been added to the revised manuscript.



(a) (b)

Fig. 9: Impacts of different forecasting methods on individual energy management. (a) Schematic diagram of edge home energy management for flexible energy resources. Edge intelligence enables smart meters to manage local energy storage, controllable household appliances, electric vehicle charging, and energy market participation based on predicted loads. (b) Comparison of the electricity cost, renewable energy accommodation ratio, and carbon emission for a non-intelligent strategy and various edge intelligent methods. The experiments were conducted on 30 buildings and houses in the BDG2 and CBTs datasets for 180 test days.

Load forecasting facilitates consumers to gain deeper insights into future energy consumption patterns, thereby supporting tailored energy management decisions. In addition to conducting accuracy analysis, we explore the impact of forecasting errors on the downstream decision-making process. Fig. 9(a) illustrates a representative home, which features distributed flexible energy resources, including solar panels, controllable appliances, electric vehicles, and energy storage systems. Note that a building can be regarded as a multi-household collective with larger-scale distributed resources. Building energy management (BEM) and home energy management (HEM) are typically achieved through two stages: 1) short-term scheduling and 2) real-time balancing. Briefly, short-term scheduling aims to minimize electricity costs while ensuring a balance between forecast demand and supply by scheduling various flexible resources for upcoming periods. To this end, the smart meter installed on each building/home first predicts the

future load using a pre-trained forecasting model and retrieves the time-of-use tariff information from the grid operator's cloud platform. On this basis, the smart meter can determine the operating strategies of storage systems and household appliances and recommend strategies for participating in the energy market. To save electricity costs, storage systems and electric vehicles can be charged during off-peak tariff periods, while grid-connected electricity sales can be conducted during peak solar generation periods. However, due to prediction errors, smart meters may require further adjustments to achieve the real-time supply and demand balance. In this case, a higher predicted load implies that consumers discard unused generated renewable energy, while a lower predicted load means that consumers have to temporarily purchase electricity in the energy market. Both situations are unfavourable for efficient energy management. In short, accurate forecasting results contribute to low additional grid electricity purchases and a high accommodation ratio of solar generation, thus reducing total carbon emissions.

We conducted comprehensive experiments on the BDG2 and CBTs datasets to showcase the effectiveness of the proposed method in enhancing decision-making for BEM and HEM. Fig. 9(b) provides a performance comparison of a non-intelligent strategy and various edge intelligent methods in terms of the electricity cost, renewable energy accommodation ratio, and carbon emission. In the non-intelligent strategy, smart meters without edge intelligence cannot provide any assistance or support for customers to schedule flexible energy resources. The results clearly show that introducing edge intelligence to smart meters can, on average, reduce electricity cost by 31.79%, increase renewable energy accommodation by 35.38%, and reduce carbon emission by 59.78% for each building. These improvements brought to each house can be found at 35.42%, 40.38%, and 49.31%, respectively. By adopting our approach, electricity cost savings of $1,176.11 per building and electricity cost savings of $18.93 per household can be expected annually. Importantly, the proposed method, which has the highest forecasting accuracy among all intelligent methods, also achieves a significant performance improvement in individual energy management. Compared to the best-performing benchmarks, our approach saves electricity cost, boosts renewable energy consumption, and reduces carbon emission for buildings and houses, reaching values of 3.08%, 1.38%, 5.42% and 2.41%, 0.76%, 1.96%, respectively. Interestingly, the prediction error is not strictly monotone with the downstream decision cost. For instance, SFLV2 outperforms FedAvg-S in residential load forecasting, but its performance in subsequent energy management is unsatisfactory.

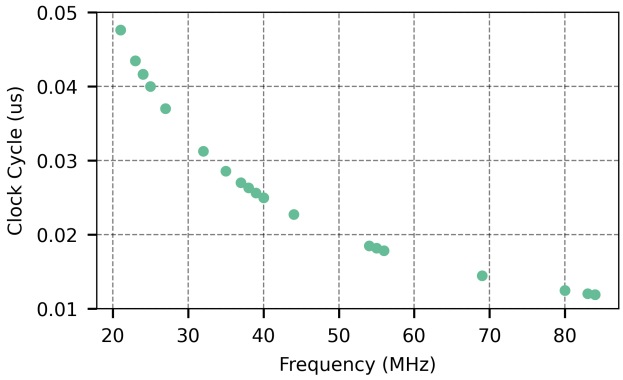
---

9. Continuing the discussion on results, the result that looks most significant is the reduction is communication cost. However, this result feels very unintuitive when thinking it over. In Fig. 5, there appears to be a massive reduction in communication cost when you compare synchronous to semi-asynchronous and asynchronous. From my understanding of the experimental setup,

the number of rounds for each experiment is fixed at 100 rounds and the experiment runs until all 100 rounds are complete. First, this choice feels a bit odd and in favor of asynchronous execution in general. A better stopping condition would be converged testing loss against the fixed test data set. However, the other issue is that the communication cost reduction doesn't make sense. In the semi-asynchronous case, the end-edge aggregation is synchronous and the edge-cloud aggregation is asynchronous. If every end-edge subtree runs for 100 rounds, then wouldn't the total training time simply be the training time of the slowest cluster of smart meters multiplied by 100? If so, then there should be no noticeable reduction in either case because the slower cluster of end devices is going to be the bottleneck that ultimately determines the total training time regardless of the other end-edge clusters continuing on asynchronously. If I'm misunderstanding, then this should be better clarified in writing.

Reply:

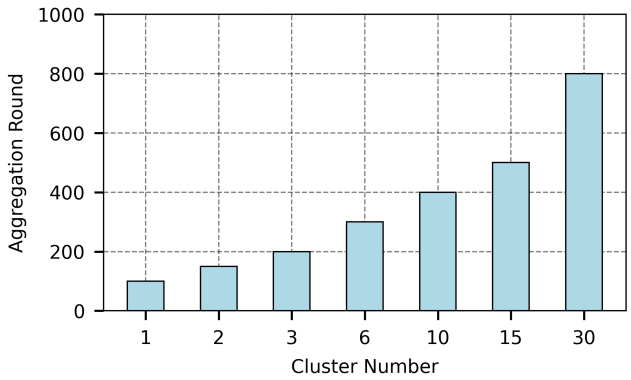
We apologize for any misunderstandings induced by incomplete details of the experimental setup. The proposed semi-asynchronous aggregation method mainly solves the delay problem brought on by large-scale heterogeneous smart meters. To simulate the device heterogeneity in a real smart grid, 30 smart meters are randomly set to different operating frequencies between 21MHz and 84MHz (see Supplementary Fig. 7).



Supplementary Fig. 7: Heterogeneous hardware configurations for 30 smart meters. To simulate the device heterogeneity in a real smart grid, 30 smart meters are randomly set to different operating frequencies between 21MHz and 84MHz, with an average frequency of 42MHz. In this setup, the computational power of the fastest smart meter is about four times the computational power of the slowest one. Note that the clock cycle is inversely proportional to the operating frequency.

In federated learning, the round is defined as the number of global model updates, that is, the number of edge-cloud aggregations in the proposed framework. In synchronous aggregation, the global training rounds for each experiment are fixed at 100 rounds, where all clusters need to upload model gradients in each round to update the global model. In asynchronous/semi-

asynchronous aggregation, only one cluster uploads model gradients in each round to update the global model, and the deviation of a single cluster gradient will make the training rounds longer for convergence. We have tried using the mentioned early stopping strategy to select the stopping round based on test loss. However, we found that the global training loss fluctuates greatly in the experiment (the loss converges stably in the local fine-tuning phase), and the early stopping strategy cannot achieve ideal results. Therefore, we manually chose the number of aggregation rounds at different numbers of clusters based on the loss of global training in 5 repeated experiments to ensure global model convergence (see Supplementary Fig. 8).



Supplementary Fig. 8: Aggregation round selected for the semi-asynchronous aggregation method. In each edge-cloud aggregation, only one cluster uploads the model gradient to update the global model in each round. We can observe that as the number of clusters increases, the deviation of individual cluster gradients makes the training rounds longer for convergence. Note that clusters with shorter training times upload model gradients more frequently than clusters with longer training times at a given time.

Note that clusters with shorter training times upload model gradients more frequently than clusters with longer training times at a given time. Consequently, not every end-edge cluster needs to run 100 rounds. To be specific, numerical calculations can be performed to estimate the benefits of communication overhead and training time owing to semi-asynchronous aggregation. Taking the number of clusters as 6 as an example, each cluster has 5 smart meters, and the aggregation rounds are set to 300.

- **Communication overhead:** In semi-asynchronous aggregation, only one cluster uploads model gradients in each round, so the total number of smart meter communication instances is 300*5 = 1500. For synchronous aggregation, all clusters need to upload model gradients in each round, so the total number of smart meter communication instances is 100*30 = 3000. Thus, the communication overhead is significantly reduced.
- **Training time**: In semi-asynchronous aggregation, the average number of aggregation rounds per cluster is 50. Since the computational power of smart meters differs by a

factor of 4, we can estimate that the shortest training time cluster has 100 aggregation rounds, while the longest training time cluster has 25 aggregation rounds. In this case, the training time of semi-asynchronous aggregation is the training time of the slowest smart meter cluster multiplied by 25, while the training time of synchronous aggregation is the training time of the slowest smart meter cluster multiplied by 100. Therefore, the training time is also significantly reduced.

To avoid misunderstandings, we have added the following details of experimental settings in the Supplementary information:

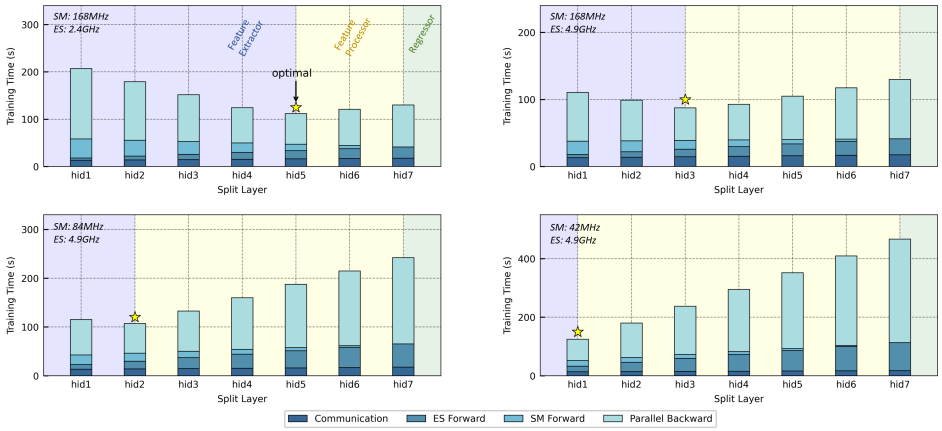
In our experiments, L2 loss is adopted for both the label loss and knowledge distillation loss. The models are trained using the Adam optimizer with an initial learning rate of 5e-4. The mini-batch size is set to 32. The weights of label loss and knowledge distillation loss are both set to 0.5. To simulate the device heterogeneity in a real smart grid, 30 smart meters are randomly set to different operating frequencies between 21MHz and 84MHz (see Supplementary Fig. 7). In synchronous aggregation, the global training rounds for each experiment are fixed at 100 rounds. In asynchronous aggregation methods, the deviation of a single cluster gradient makes the training rounds longer for convergence, so we manually chose the number of training rounds at different numbers of clusters based on the loss of global training to ensure global model convergence (see Supplementary Fig. 8). Note that a local fine-tuning personalization strategy is incorporated into all federated-based methods for 30 rounds. Each experiment is repeated 5 times to eliminate the effect of randomness.

---

10. Regarding the method for finding the optimal model split, it would be interesting to see this solution run on larger models—even if it means ignoring the memory constraints of the MCU nodes. This is generally just an interesting research problem in split learning. But, the fact the authors only focus on very small models with 3-5 hidden layers, it becomes exceedingly difficult to be convinced that the solution is able to make a good decision. For instance, I imagine that in the case of 3 hidden layers, a naïve random solution has a 33% chance of making the optimal model split.

Reply: Thank you for the suggestion to evaluate our method for finding the optimal model split on larger models. We understand the importance of demonstrating the effectiveness and scalability of our solution in handling more complex models.

We have extended our experiments to include larger models with deeper 7 hidden layers while maintaining the focus on the specific research problem in split learning. The additional results presented in Supplementary Fig. 5 show that the efficiency-optimal split strategy effectively finds the split layer to minimize the training time for four distinct hardware configurations. Splitting the layer granularity based on the proposed method can significantly improve latency



Supplementary Fig. 5: Effectiveness of the efficiency-optimal model splitting strategy considering deeper neural network with 7 hidden layers. Each hidden layer is considered a candidate split layer. The split layers that yield the best efficiency are annotated. The hidden layers contained in the feature extractor, feature processor, and regressor after the optimal splitting are indicated with different colors. The total training times under four distinct hardware configurations when choosing different split layers are provided. The stacked histograms represent the measured times for communication, forward propagation of the edge server and smart meter, and parallel backward propagation, arranged from bottom to top.

and efficiency.

---

11. The decision to cluster the smart meters by compute power using a (balanced) K-means algorithm feels very unintuitive. In hierarchical FL settings, the hierarchy generally is established based on the true underlying network. For instance, in a paper by Hudson et al. on FL for the nonintrusive load monitoring problem, they consider 3-tier federated aggregation of smart metering data where the middle aggregator (analogous to the edge cloud in the authors' submitted manuscript) is localized to a neighborhood in the neighborhood area network in the AMI system and it is connected to home area networks (each with a smart meter) most local to it. This seems more natural for hierarchical networks. Additionally, as the authors have mentioned, smart meters are notoriously low-power in terms of compute availability. Clustering them based on compute capacity seems less important since they likely do not vary that widely anyway. Clustering them based on geographic distance is more intuitive from a networks perspective. Finally, it is more logical to apply this approach due to the naturally non-iid data distributions of energy consumption across different neighborhoods. For instance, more affluent neighborhoods are likely going to have homes with different energy consumption patterns than homes in a poor neighborhood.

Reply:

Thank you for the comment on the adoption of clustering approaches. In the proposed end-edge-cloud framework, the primary objective of clustering is to ensure that the training time for all smart meters managed by a single server is similar, thereby avoiding delays in intra-cluster aggregation. Based on the analysis in equation (3), the total computation time is related to three physical hardware configurations, i.e., computational power of edge server P_{es}, computational power of smart meter P_{sm}, and communication rate R. Since the computational power of the edge server is the same for all smart meters within the same cluster, we consider $[P_{sm}, R]$ as the feature vector in the clustering algorithm. Here we would like to clarify that smart meters installed in different periods and regions may possess varying computational power due to the development of technical standards, product aging, changes in market demand, and so on.

$$\begin{aligned} T(\alpha) = & \frac{3s|D| + 2\alpha|\mathbf{w}|}{R} + \frac{\alpha\beta n|D||\mathbf{w}|}{P_{sm}} + \frac{(1-\alpha)\beta n|D||\mathbf{w}|K}{P_{es}} \\ & + \max\left\{\frac{\alpha(1-\beta)n|D||\mathbf{w}|}{P_{sm}}, \frac{(1-\alpha)(1-\beta)n|D||\mathbf{w}|K}{P_{es}}\right\} \end{aligned} \tag{3}$$

Furthermore, we can also find in equation (3) that the training time is positively correlated to the number of smart meters K within the cluster. In this case, traditional clustering methods may show a significant imbalance in the number of meters within different clusters due to the effect of extreme values. The adopted balanced K-means clustering approach can make the number of smart meters within each cluster closely approximated to each other, thus avoiding the frequent or infrequent gradient uploads of a particular cluster in edge-cloud asynchronous aggregation due to its excessively long or short completion time.

Besides, we think that the non-iid problem is not important in the proposed method. Specifically, the training completion time varies across clusters after clustering, so the model gradient of a fast-trained cluster is utilized more frequently to update the global model. Suppose consumers with the same electricity consumption pattern are designated into the same cluster, meaning that different clusters have non-iid data distribution. The imbalanced update frequency may exacerbate the impact of the heterogeneous data distribution. For example, the final global model will tend to perform better on the fastest trained cluster rather than performing well across all clusters.

In short, we kindly believe that the adopted balanced K-means clustering method is quite well-suited to our proposed hierarchical framework. However, for large-scale smart meters, geographical space is indeed very important considering the additional overhead brought by cross-regional communication. So we added the paper by Hudson et al. as a relevant reference to the literature review section in the revised manuscript.

---

12. In Table 2, FedAvg-M is the next best algorithm based on the authors' metrics. I would be

curious to see how the FedProx algorithm performs against the authors' proposed solution. This federated aggregation algorithm adds a proximal term (based on the norm between the global model and the locally-trained model) to the loss before backprop. This "grounds" the locally-trained models to not stray too far apart from the global model and has shown well to work on non-iid data. It might make for a more apt comparison to the knowledge distillation-driven approach proposed by the authors.

Reply:

Thank you for the suggestion on comparing the performance of relevant SOTA methods. We have added the FedProx algorithm as a benchmark in our extensive experiments to illustrate the superiority of the proposed method. The visualization results discussion are given as follows.

Table 1: Performance of different methods on BDG2 and CBTs in terms of accuracy, memory, training time, and communication overhead per round.

Method	BDG2			CBTs			Memory (KB)	Training Time (s)	Communication (KB)
	RMSE	MAPE	MAE	RMSE	MAPE	MAE			
Cen	22.82	8.41	8.20	0.4780	26.98	0.2727	2578.42 (1.0×)	586.42 (2.05×)	-
Local-S	23.56	8.13	8.12	0.4698	28.19	0.2726	152.53 (16.9×)	41.19 (29.14×)	-
FedAvg-S	22.79	7.67	7.98	0.4681	27.04	0.2699	152.53 (16.9×)	46.49 (28.80×)	5.50 (27.18×)
FedProx-S	22.58	7.71	7.91	0.4668	27.17	0.2678	152.53 (16.9×)	46.49 (28.80×)	5.50 (27.18×)
Local-M	22.84	7.75	8.03	0.4645	26.67	0.2682	2578.42 (1.0×)	1187.15 (1.01×)	-
FedAvg-M	22.25	**6.96**	7.48	**0.4614**	25.81	0.2637	2578.42 (1.0×)	1200.44 (1.0×)	149.50 (1.0×)
FedProx-M	22.21	7.16	7.62	0.4622	25.76	0.2639	2578.42 (1.0×)	1200.44 (1.0×)	149.50 (1.0×)
Split	23.27	7.68	7.96	0.4679	26.83	0.2687	103.75 (24.8×)	96.00 (12.50×)	91.25 (1.63×)
SFLV1	22.34	7.25	7.68	0.4647	26.03	0.2664	103.75 (24.8×)	96.49 (12.44×)	96.75 (1.54×)
SFLV2	22.76	7.53	7.89	0.4674	26.49	0.2683	103.75 (24.8×)	96.49 (12.44×)	96.75 (1.54×)
Proposed	**22.17**	6.98	**7.44**	0.4630	**25.74**	**0.2636**	115.07 (22.4×)	62.41 (19.23×)	74.43 (2.01×)

The results in Table 1 indicate that **beyond the same efficiency metrics, the improved FedProx and FedAvg algorithms do not differ significantly in terms of accuracy performance.** This suggests that the overall data distribution among consumers is not significantly heterogeneous, so the proximal term introduced by FedProx has a limited effect on improving model performance. We further find that FedProx no longer performs as well as FedAvg when using the deeper model. This may be due to the constraint of the proximal term in model updates leading to underfitting phenomena in large-scale models (see Supplementary Table 3).

Besides, as demonstrated in Fig. 5 and Fig. 6, **FedProx performs better on the BDG2 dataset, while FedAvg performs better on the CBTs dataset.** This reflects that the electricity consumption patterns of different buildings in the BDG2 dataset exhibit stronger non-iid characteristics.

In summary, it can be concluded that **the proposed method remains the best-performing of**

Supplementary Table 3: Performance of different methods on BDG2 and CBTs in terms of accuracy, memory, training time, and communication overhead per round considering deeper neural network with 7 hidden layers.

Method	BDG2			CBTs			Memory (KB)	Training Time (s)	Communication (KB)
	RMSE	MAPE	MAE	RMSE	MAPE	MAE			
Local-M	22.75	7.68	8.03	0.4678	26.68	0.2721	5178.25 (1.0x)	4335.86 (1.01x)	-
FedAvg-M	22.33	**6.84**	7.44	**0.4617**	26.28	0.2638	5178.25 (1.0x)	4387.33 (1.0x)	761.5 (1.0x)
FedProx-M	22.42	7.01	7.52	0.4620	26.17	0.2641	5178.25 (1.0x)	4387.33 (1.0x)	761.5 (1.0x)
Split	22.93	7.89	8.14	0.4694	26.74	0.2713	103.75 (49.9x)	281.86 (15.55x)	91.25 (8.34x)
SFLV1	22.56	7.03	7.57	0.4963	26.43	0.2650	103.75 (49.9x)	282.84 (15.51x)	96.75 (7.87x)
SFLV2	22.75	7.18	7.66	0.4647	26.39	0.2663	103.75 (49.9x)	282.84 (15.51x)	96.75 (7.87x)
Proposed	**22.22**	6.86	**7.38**	0.4626	**26.12**	**0.2637**	115.07 (45x)	214.68 (20.43x)	74.43 (10.23x)

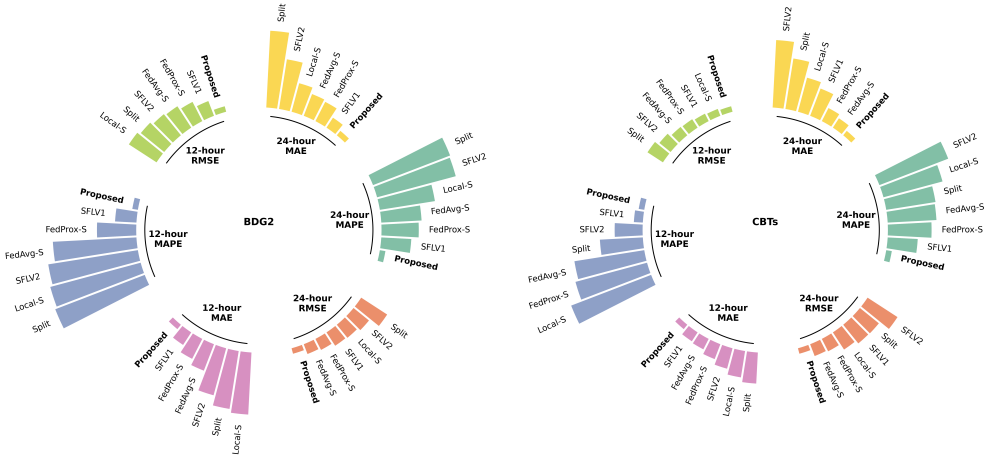


Fig. 7: Performance evaluation of the proposed model for different forecasting ranges. We present the accuracy improvement of our method compared with other device-friendly methods for 12-hour-ahead and 24-hour-ahead forecasting.

all the on-device feasible methods on both datasets.

---

*13. Plots need to be clearer. More specifically, plots often do not have clear titles or subtitles to clarify what they are communicating. One such example is Figure 3(b). The individual subplots in this figure are not clearly labeled so it is unclear what distinguishes these 4 subplots from one another. Another comment on clear plotting can be said about the choice to use "split ratio" in Figure 3(a). This is a very inaccessible metric (*especially* without the supplemental text). It might be clearer to just mention the layer(s) that split and placed on the edge server—this seems to be indicated by the color, but the ratio is a confusing detail.*

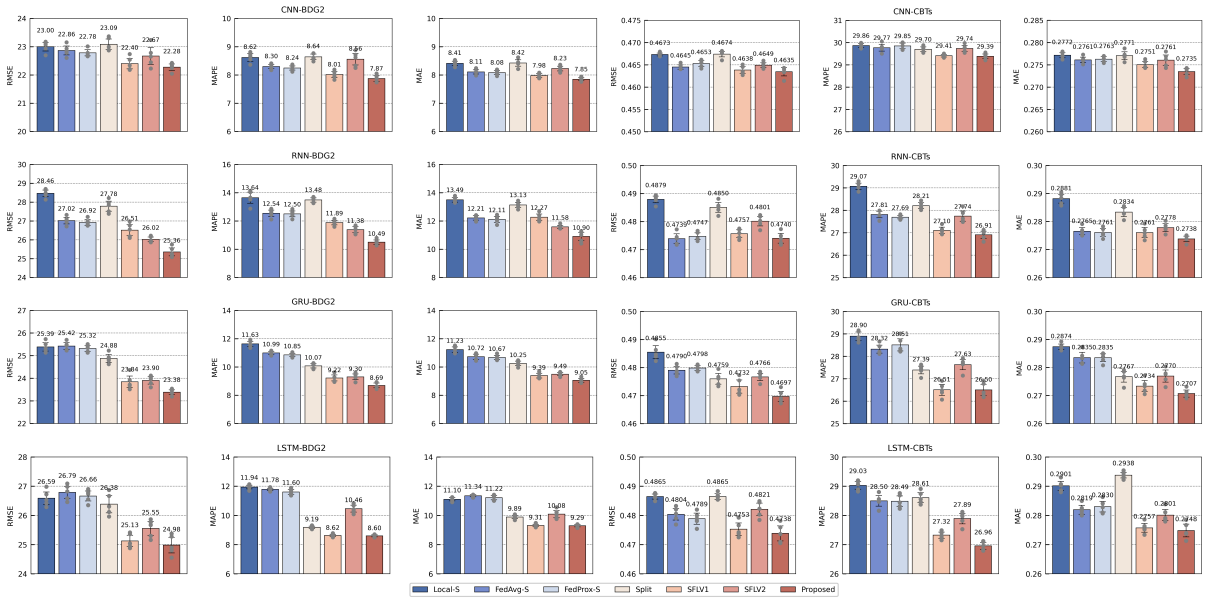


Fig. 8: Performance evaluation of the proposed method with different neural networks as the backbone. We compare the accuracy metrics of our method with other device-friendly methods with a CNN, RNN, GRU or LSTM as the backbone. The mean accuracy with 95% confidence intervals is presented with 5 independent experiments.

Reply: Thank you for the valuable comment on plot clarity. we acknowledge the reviewer's concerns and appreciate the constructive feedback. To address this issue, we have added concise details to indicate the variables and conditions represented.

As shown in Fig. 4, we have added hardware configuration labels to each subplot, identifying the distinct characteristics and differences between them. In addition, we have added a legend explaining the training process corresponding to each training time color block of the stacked histograms. Regarding the use of "split ratio", we understand that it may not be the most accessible metric, especially without supplemental text. We have revised this plot by providing a more straightforward representation of the split layers (hidden layers 1...5). The hidden layers contained in the feature extractor, feature processor, and regressor after the optimal split are indicated with different colors. This will enable readers to more easily grasp the intended purpose and conclusions drawn from this figure.

---

14. The authors state that this is the first work to consider edge intelligence for smart meters. This is not true and other relevant works should be adequately highlighted—with the authors providing a clear distinction between their work and what's already been done. Below are some examples of papers exploring this topic:

- Fekri, Mohammad Navid, Katarina Grolinger, and Syed Mir. "Distributed load forecasting using smart meter data: Federated learning with Recurrent Neural Networks." International Journal

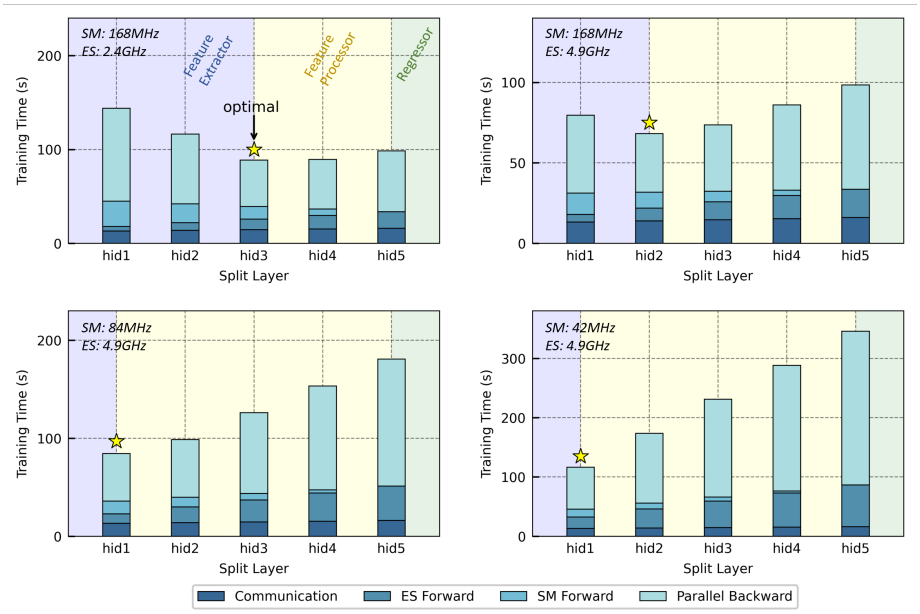


Fig. 4: Effectiveness of the efficiency-optimal model splitting strategy. Each hidden layer is considered a candidate split layer. The split layers that yield the best efficiency are annotated. The hidden layers contained in the feature extractor, feature processor, and regressor after the optimal splitting are indicated with different colors. The total training times under four distinct hardware configurations when choosing different split layers are provided. The stacked histograms represent the measured times for communication, forward propagation of the edge server and smart meter, and parallel backward propagation, arranged from bottom to top.

of Electrical Power& Energy Systems, 137 (2022): 107669.

- Fekri, Mohammad Navid, Katarina Grolinger, and Syed Mir. "Asynchronous adaptive federated learning for distributed load forecasting with smart meter data." International Journal of Electrical Power& Energy Systems, 153 (2023): 109285.

- Hudson, Nathaniel, Md Jakir Hossain, Minoo Hosseinzadeh, et al. "A framework for edge intelligent smart distribution grids via federated learning." 2021 International Conference on Computer Communications and Networks (ICCCN), IEEE, 2021.

- Wang, Yi, Ning Gao, and Gabriela Hug. "Personalized federated learning for individual consumer load forecasting." CSEE Journal of Power and Energy Systems, 9.1 (2022): 326-330.

- Taïk, Afaf, and Soumaya Cherkaoui. "Electrical load forecasting using edge computing and federated learning." ICC 2020-2020 IEEE international conference on communications (ICC), IEEE, 2020.

Reply:

Thank you for suggesting relevant references. We agree with the reviewer that edge intelli-

gence has been investigated by some research. However, these studies mainly utilize the smart meter data to carry out simulation experiments instead of truly implementing their methods on smart meter hardware. The main point that differentiates our work from the existing work is that we have solved the resource-constrained problem of smart meters and showcased the effectiveness of the proposed method on a hardware platform, which is the first attempt to realize intelligence on the smart meters from theory to practice to best of our knowledge.

We have enriched the literature review to highlight relevant work using smart meter data for edge intelligence and provided a clearer distinction between our work and existing work.

Several studies have investigated FL for edge intelligence, such as [1-5]. However, these studies mainly utilize smart meter data to carry out simulation experiments instead of implementing their methods on resource-constrained smart meter hardware. There is still a lack of a unified framework that considers all perspectives of model accuracy, on-device memory footprint, computation speed, and communication overhead to fully achieve on-device intelligence.

While previous studies have often overlooked the limitations of the communication and computational capabilities of edge devices, our work addresses the challenge of translating these methods into practical, real-world applications tailored for smart meter hardware. We focus on overcoming the constraints inherent to edge environments, ensuring that our solutions are not only theoretically sound but also viable for on-the-ground deployment.

[1] Fekri, Mohammad Navid, Katarina Grolinger, and Syed Mir. "Distributed load forecasting using smart meter data: Federated learning with Recurrent Neural Networks." International Journal of Electrical Power& Energy Systems, 137 (2022): 107669.

[2] Fekri, Mohammad Navid, Katarina Grolinger, and Syed Mir. "Asynchronous adaptive federated learning for distributed load forecasting with smart meter data." International Journal of Electrical Power& Energy Systems, 153 (2023): 109285.

[3] Hudson, Nathaniel, Md Jakir Hossain, Minoo Hosseinzadeh, et al. "A framework for edge intelligent smart distribution grids via federated learning." 2021 International Conference on Computer Communications and Networks (ICCCN), IEEE, 2021.

[4] Wang, Yi, Ning Gao, and Gabriela Hug. "Personalized federated learning for individual consumer load forecasting." CSEE Journal of Power and Energy Systems, 9.1 (2022): 326-330.

[5] Taïk, Afaf, and Soumaya Cherkaoui. "Electrical load forecasting using edge computing and federated learning." ICC 2020-2020 IEEE international conference on communications (ICC), IEEE, 2020.

Reviewer 1 (Remarks on code availability):

15. In all honesty, a very significant portion of their code relies on a microcontroller framework I am not at all familiar with. So it is difficult for me to comment much on the reproducibility of their code from that perspective. As for the simulation parts that do not rely on this framework, the codes seem reproducible enough. However, the simulation tests appear to be simplistic enough that they can be easily adapted or rewritten by other members of the academic community. The simulation code for this work is very custom-fitted for this authors specific use case and does not appear to be extensible.

However, they may be potential that the microcontroller codes could have some reach and reusability, but I cannot say much on that part of their code.

Reply:

Thank you for your valuable feedback on the reproducibility and extensibility of our code. Regarding the **microcontroller framework**, we would like to clarify that our hardware code strictly adheres to the underlying architecture. All the codes are based on fundamental algorithms such as array operations and for-loops, which ensures strong portability. We understand your concerns about the unfamiliarity with the framework, and we will strive to provide better documentation and support to make it more accessible to other researchers. As for the **simulation codes**, we have carefully taken your feedback into account and made improvements to enhance their extensibility. We have added more comments and annotations to the code to provide better understanding and ease of use. Furthermore, we have updated the GitHub website (`https://github.com/SimonLi2000/Make-Smart-Meter-Really-Smart`) with more detailed usage instructions and documentation to assist users in adapting and extending the simulation code.

Reviewer 2 (Remarks to the Author):

In this paper, the authors investigated the potential of smart meters for supporting the excavation of demand-side flexibility by using edge intelligence. Generally, the data received by a single smart meter is limited while sharing data directly in the network causes the leakage of the data, introducing privacy issues. From this perspective, the authors leveraged federated learning. In this way, only the model parameters would be shared, instead of the original data, improving data privacy. Considering the resource limitations due to the physical size of smart meters, the authors decided to seek help from the edge server to perform local training via split learning. In this way, the smart meter only executed partial model layers and delegated the heavy training processes to the nearby edge server. Combining split learning and federated learning, the authors proposed the end-edge-cloud federated split learning framework to achieve smart grids. In addition, the authors implemented a hardware platform to evaluate their approach with other representative approaches.

As a researcher in edge intelligence, I am really happy to see the implementation of edge intelligence in the industry and also appreciate the effort of the authors in implementing the hardware platform. Such study is needed but there are various places that require clarification and further consideration to make the article more convincing and comprehensive.

Reply:

Thank you for the careful reading of our work and for acknowledging its interest and solidity. We will now respond point-by-point to your comments and questions.

---

1. While split learning (SL) is in general efficient in reducing overall training time by delegating heavy training computation on powerful edge servers (compared to smart meter), the training time on edge servers is not further analysed. Offloading too many training tasks to edge servers may introduce an extra bottleneck in the system. To eliminate such risk, the authors are suggested to add the overhead analysis to figure this out and explicitly demonstrate it to the readers. If it becomes an issue, I suggest using pipeline-based approaches to schedule learning tasks on smart meters and edge servers in a more flexible way.

Reply:

Thanks for your valuable suggestion on model training task offloading. Admittedly, although edge servers are powerful relative to smart meters, each server needs to take on the heavy training computation of a cluster of smart meters. The choice of split ratio encounters a dilemma: offloading excessive training tasks to the edge servers may result in significant bottlenecks in training time; however, offloading excessive training tasks to the smart meters may lead to

memory overflow. To mitigate such risk, we aim to find an optimal split ratio α^* that minimizes the overall training time T subject to memory constraints M_{sm} of smart meters, which can be formulated as an optimization problem (1).

$$\begin{aligned} &\min_{\alpha} T(\alpha) \\ &\text{s.t. } M(\alpha) \leq M_{sm} \end{aligned} \tag{1}$$

Consequently, we analyze the peak memory footprint of key elements in equation (2), including model memory, intermediate memory, and optimizer memory.

$$M(\alpha) = 32 \times \sum_{i=1}^{\lfloor \alpha L \rfloor} |B| \left(|\mathbf{w}_i| + 2 |\mathbf{a}_i| \right) + 3 |\mathbf{w}_i| \tag{2}$$

In addition, we analyze the training time of the main processes in Eq. (3), including computation in edge servers and smart meters and communication between edge servers and smart meters. The details of overhead analysis for training time on edge servers are provided in the revised manuscript.

First, we analyze the time spent on computation including forward and backward propagation. The amount of computation required for each parameter is assumed to be equal and is denoted as n. Since $|\mathbf{w}|$ is typically much larger than $|\mathbf{a}|$ in (2), the computational complexity of the complete model training can be represented as $\mathcal{O}(|D|\,|\mathbf{w}|)$ for the entire dataset. Therefore, the total amount of computation in the training process can be characterized as $n|D||\mathbf{w}|$. Let β denote the fraction of computation used for forward propagation. Initially, smart meters concurrently perform forward propagation on their local models, which takes time of $\frac{\alpha\beta n|D||\mathbf{w}|}{P_{sm}}$. Similarly, the edge server then carries out forward propagation on the edge-side models for each smart meter, which takes time of $\frac{(1-\alpha)\beta n|D||\mathbf{w}|K}{P_{es}}$. In our training method detailed later, the smart meters and edge server backpropagate their respective models in parallel. The parallel propagation time is determined by the maximum value for the model of the smart meters and edge server, which can be expressed as $\max\{\frac{\alpha(1-\beta)n|D||\mathbf{w}|}{P_{sm}}, \frac{(1-\alpha)(1-\beta)n|D||\mathbf{w}|K}{P_{es}}\}$.

Second, we analyze the time spent on communication. In each round, the smart meters communicate with the edge server to upload and download the weights of the end-side model, with each process requiring time of $\frac{\alpha|\mathbf{w}|}{R}$. Since the complete model is split into three parts, the intermediate activations of the split layers need to be transmitted twice between the smart meters and edge server for forward propagation, which will take time of $\frac{2s|D|}{R}$. In our method detailed later, the edge server no longer returns the gradient of the split layer to the smart meters. Thus, the smart meters send the gradients of the activations of the split layer back to the edge server, taking time of $\frac{s|D|}{R}$.

In summary, the overall training time $T(\alpha)$ used per round can be formulated as:

$$T(\alpha) = \frac{3s|D| + 2\alpha|\mathbf{w}|}{R} + \frac{\alpha\beta n|D||\mathbf{w}|}{P_{sm}} + \frac{(1-\alpha)\beta n|D||\mathbf{w}|K}{P_{es}} + \max\left\{\frac{\alpha(1-\beta)n|D||\mathbf{w}|}{P_{sm}}, \frac{(1-\alpha)(1-\beta)n|D||\mathbf{w}|K}{P_{es}}\right\} \tag{3}$$

Ultimately, optimization problem (1) can be solved by piecewise dissection. Theorem 1 offers guidance for determining the efficiency-optimal split ratio α^*. The proof is provided as Supplementary Proof. 1.

Theorem 1 (Efficiency-optimal split ratio) *If* $P_{es} > K\left(\frac{1}{nR|D|} + \frac{\beta}{P_{sm}}\right)^{-1}$, *we have*

$$\alpha^* = \alpha_{upper} \tag{4}$$

If $P_{es} < \beta K\left(\frac{2}{nR|D|} + \frac{1}{P_{sm}}\right)^{-1}$, *we have*

$$\alpha^* = \alpha_{lower} \tag{5}$$

If $\beta K\left(\frac{2}{nR|D|} + \frac{1}{P_{sm}}\right)^{-1} \leq P_{es} \leq K\left(\frac{1}{nR|D|} + \frac{\beta}{P_{sm}}\right)^{-1}$, *we have*

$$\alpha^* = \begin{cases} \alpha_{upper} & \text{if } \left(\frac{P_{es}}{KP_{sm}} + 1\right)^{-1} \geq \alpha_{upper} \\ \left(\frac{P_{es}}{KP_{sm}} + 1\right)^{-1} & \text{if } \alpha_{upper} \leq \left(\frac{P_{es}}{KP_{sm}} + 1\right)^{-1} \leq \alpha_{lower} \\ \alpha_{lower} & \text{if } \left(\frac{P_{es}}{KP_{sm}} + 1\right)^{-1} \leq \alpha_{lower} \end{cases} \tag{6}$$

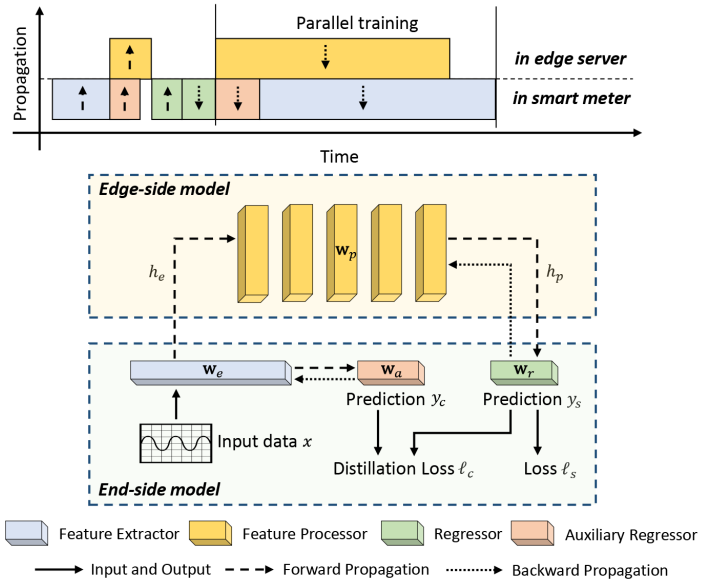
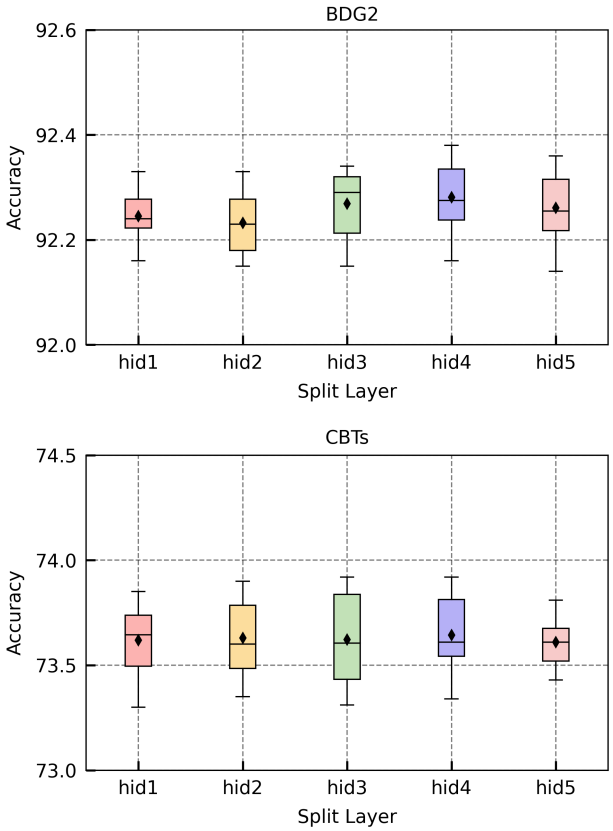


Fig. 10: Illustration of the process and pipeline in collaborative model training.

Besides, in traditional split learning, entire model training tasks are executed sequentially between smart meters and edge servers. In this case, the smart meters are idle and cannot update the local model before receiving the gradients returned by the edge server. The **pipeline** of the proposed method is shown in Fig. 10, where smart meters and edge servers can execute model training in parallel. This novel strategy greatly improves computational efficiency and computational resource utilization in a flexible manner.

---

2. Another concern is still about SL. The authors proposed a ratio to determine the way to split the model to minimize the overall training time while fulfilling the memory constraints. However, the layers selected based on the ratio are not explicitly discussed. In many machine learning models, the characteristics of different layers are different. The layers closer to the input are more important to the feature extraction and the layers closer to the output are more important to the feature fusion and integration. For example, in the case that three of the seven layers are put on the smart meter, what's the performance of (1, 2, 7), (1, 4, 7), (1, 6, 7) or others? More evidence could be provided to enhance the feasibility of this approach in real-world scenarios.



Supplementary Fig.4: Comparison of forecasting accuracy when splitting at different hidden layers. The experimental results reveal that splitting at different hidden layers does not significantly affect the forecasting accuracy of the model on both datasets.

Reply:

Thanks for your constructive comment on the selection of split layers. Specifically, each hidden layer is considered as a candidate split layer. The optimal split layer can be determined in terms of both accuracy and efficiency.

- The proposed method aims to maximize the efficiency of distributed computation for model training. The proposed method requires two split layers to split the model into three components. This is because multiple splits (such as 1, 4, 7 on the smart meter) in the middle of the model compared to two splits (such as 1, 2, 7, or 1, 6, 7 on the smart meter) are generally considered inefficient. For instance, training the entire model with multiple splits requires a higher volume of communications ($1 \rightarrow 2, 3 \rightarrow 4 \rightarrow 5, 6 \rightarrow 7$ need 4 times) compared to two splits ($1, 2 \rightarrow 3, 4, 5, 6 \rightarrow 7$ need 2 times). If only three layers can be assigned to the smart meter, we prioritize the hidden layers close to the input and output layers as the split point.
- We recognize that the relationship between model accuracy and model split layers varies from dataset to dataset and cannot be directly analyzed and quantified. We have conducted experiments on forecasting accuracy versus split layers, where the results in Supplementary Fig. 4 show that splitting the model at different layers does not significantly affect the accuracy of the whole model. This may be attributed to that different split points only mean that the components of an entire model are deployed on different devices for training without significantly changing model convergence. Therefore, similar accuracy performance can be achieved when allocating more hidden layers close to the input layer (such as 1, 2, 7 on the smart meter) or allocating more hidden layers close to the output layer (such as 1, 6, 7 on the smart meter) on the smart meter.

Based on the above analysis, we can conclude the proposed method requires two split layers to split the model into three components. One feasible strategy of split layer selection could be that: the subsequent split point is fixed as the last hidden layer; we only discuss how to determine the previous split layer. Theorem 1 has analyzed the efficiency-optimal split ratio of the model in different cases. Hence, We can obtain the optimal split layer by moving the previous split point until the end-side model size is closest to the optimal split ratio. We report the total training times under four distinct configurations of computation power of the edge servers and smart meters in Fig. 4. We can observe that the proposed efficiency-optimal split ratio can serve as a guideline for how to split the model to minimize the training times. By adopting the proposed split strategy, the allocation of the complete model can benefit from up to 2.97$\times$ shorter training times.

---

3. As mentioned by the authors (also well acknowledged in the community), edge devices (in-

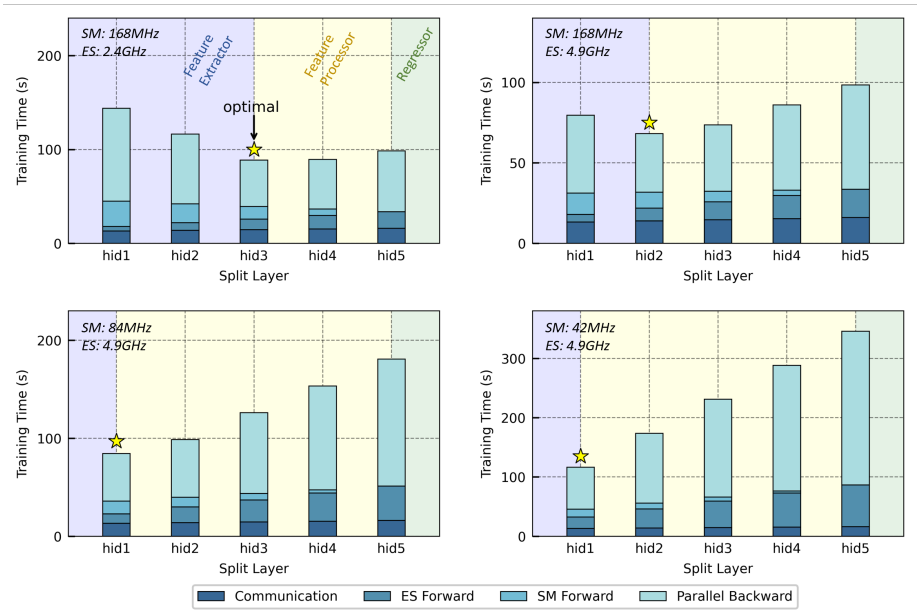


Fig. 4: Effectiveness of the efficiency-optimal model splitting strategy. Each hidden layer is considered a candidate split layer. The split layers that yield the best efficiency are annotated. The hidden layers contained in the feature extractor, feature processor, and regressor after the optimal splitting are indicated with different colors. The total training times under four distinct hardware configurations when choosing different split layers are provided. The stacked histograms represent the measured times for communication, forward propagation of the edge server and smart meter, and parallel backward propagation, arranged from bottom to top.

cluding smart meters) are also resource-constrained. In this work, the authors used the meter with only 192KB of SRAM. To analyse the memory limitation of the smart meter, the authors mentioned several types of memory usage in Section 4.1. However, the storage usage of training data is not mentioned. How much memory will those data occupy? If the data needs more storage resources, will the model training be impacted? Since this is an online training framework, more data will be collected. Will this make the memory issue more serious?

Reply:

Thanks for your comment on the memory usage of training data. As mentioned in the experimental setup, the training set consists of hourly data for an entire year. Each time slot contains five features, including load consumption and calendar information (month, day, weekday, hour). Each data point is stored as a 32-bit float type. **Therefore, the total memory usage for the training data can be calculated as follows: 365*24*5*4B = 175.2KB.**

However, the data and learning-related parameters are typically stored in different types of memory space in hardware. The constants, such as preloaded datasets, can be stored in nonvolatile FLASH memory. The variables, such as the weights and gradients involved in model

training, are cached in volatile SRAM memory because they are frequently read and written during the training process. In brief, the memory required for historical data storage will not have an impact on model training.

The smart meter collects a steady stream of newly generated load consumption data. **Here we can provide two options for updating the model with new data.** The first approach is to retrain the model by offline learning. Typically, smart meters have 1GB of FLASH memory space. According to the above calculations, it can store more than 5 years of historical data, which is adequate to train a model. The second approach is to fine-tune the model dynamically through online learning. The smart meter only needs to store real-time data instead of the entire dataset.

---

4. The last point is about privacy protection via federated learning. However, this is more related to computer science rather than engineering. In edge intelligence, there exists a number of studies that successfully infer partial data based on model parameters. Thus, the paper can be more concise to say enhancing privacy (the authors did this in several places) but not be too confident in data protection.

Reply:

Thanks for your insightful suggestion on word conciseness. We agree with you that privacy protection via federated learning is typically a statement in computer science rather than engineering. Even though the raw data is not shared directly among participants, the attacker may extract some information from the shared model parameters, potentially compromising privacy. We have revised the statement 'privacy-preserving' to 'privacy-enhancing' in the revised manuscript. We believe that this adjustment will improve the overall quality and accuracy of our work.

---

5. There are also some written issues to be fixed such as missing descriptions of notations and abbreviations.

Reply:

Thank you for pointing out the written issues in this paper. We apologize for any confusion or inconvenience caused by these omissions. We have carefully revised the manuscript to ensure that all notations and abbreviations are clearly defined and explained.

Reviewer 3 (Remarks to the Author):

The paper provides an end-edge-cloud federated split learning framework model to enable training on resource-constrained smart meters. Imho, the work is more like an algorithm improvement instead of a widely impacted study. It may be a good technical paper, but the work done is not in the style of Natural Communication research.

Reply:

While we hold the reviewer's perspective in the highest regard, we wish to claim that our study represents a **pioneering effort in the realm of smart meter-based edge intelligence within the context of smart grids**, which can achieve a wide impact and serve broad interests from following perspectives:

- Our research provides a feasible and efficient approach to **harnessing the existing ubiquitous smart meters without the need for additional investment in computational facilities**. Besides, the problems of heavy communication burden, limited hardware resources, and device heterogeneity in the large-scale smart meter system are well resolved, making our method practicable in the real-world scenario. The experiments on the hardware platform are conducted with 200 smart meters, which shows the effectiveness and applicability of our approach in real-world settings.
- Our research is not limited to on-device load forecasting but **provides new directions for broader edge intelligence applications** in the smart grid, such as on-device monitoring and on-device control. This will help consumers better exploit flexible resources to save costs and accommodate more distributed renewable energy, and also facilitate distribution system operators to better observe the system's status and manage the system to lower operation costs and improve the reliability of the energy supply.
- Our research **enables the utilization of distributed big data in a privacy-enhancing manner**, which will increase consumers' willingness for smart meter adoption, thus promoting smart meter penetration and contributing to the digitalization and consequent decarbonization of smart grids.
- We investigate the impact of the proposed on-device forecasting approach on individual household energy management and demonstrate its great effectiveness in **reducing electricity cost (31.79%), promoting renewable energy accommodation (35.38%), and lowering carbon emissions (59.78%).**

With the above considerations, we respectfully hope the reviewer to see the wide impact of this study.

We have carefully checked the aims and scope of *Nature Communications* on the website that 'Nature Communications is an open access, multidisciplinary journal dedicated to publishing high-quality research in all areas of the biological, health, physical, chemical, Earth, social, mathematical, applied, and engineering sciences. Papers published by the journal aim to represent important advances of significance to specialists within each field' (`https://www.nature.com/ncomms/aims`). We believe this study is an applied engineering science that is within the aims and scope of *Nature Communications*. We have also referred to a series of high-quality publications in *Nature Communications* about federated learning that include lots of technical content [1-3]. Additionally, recent publications in *Nature Communications* have covered topics related to smart grids [4-6], which highlight the relevance and growing interest in smart grid research within the journal. We reckon our paper is also in the interests and style of the journal considering its merits of wide impact and technical novelty.

[1] Yang H, Lam K Y, Xiao L, et al. Lead federated neuromorphic learning for wireless edge artificial intelligence[J]. Nature Communications, 2022, 13(1): 4269.

[2] Wu C, Wu F, Lyu L, et al. A federated graph neural network framework for privacy-preserving personalization[J]. Nature Communications, 2022, 13(1): 3091.

[3] Kalra S, Wen J, Cresswell J C, et al. Decentralized federated learning through proxy model sharing[J]. Nature Communications, 2023, 14(1): 2899.

[4] Jacob R A, Paul S, Chowdhury S, et al. Real-time outage management in active distribution networks using reinforcement learning over graphs[J]. Nature Communications, 2024, 15(1): 4766.

[5] Sun Q, Ma H, Zhao T, et al. Break down the decentralization-security-privacy trilemma in management of distributed energy systems[J]. Nature Communications, 2024, 15(1): 4508.

[6] Wang R, Ji H, Li P, et al. Multi-resource dynamic coordinated planning of flexible distribution network[J]. Nature Communications, 2024, 15(1): 4576.

Strength

1. The paper gives a detailed consideration of the intellectualization of smart meters.

2. It is commendable that the work provides hardware platform validation of the proposed method.

Reply:

Thank you for the careful reading of our work and for acknowledging its interest and solidity.

Weakness

1. The concepts, e.g. federated learning, and edge intelligence, are not new. Distributed learning/training structure and privacy concerns have been extensively studied in smart grid and communication fields.

Reply:

We sincerely respect the reviewer's opinion, but we would like to claim the difference between our work and existing studies and also the contributions of this paper. We agree with the reviewer that federated learning-based edge intelligence has been investigated by some research. However, these studies mainly utilize the **smart meter data** to carry out simulation experiments instead of truly implementing their methods on smart meter hardware. The main point that differentiates our work from the existing work is that we have solved the resource-constrained problem of smart meters and showcased the effectiveness of the proposed method on a hardware platform, which is **the first attempt to realize intelligence on the smart meters from theory to practice to best of our knowledge**.

The contributions and novelties of this work are three-fold:

Firstly, **we investigate a novel problem to achieve on-device intelligence from a holistic perspective**, where privacy enhancement, model accuracy, on-device memory footprint, computation speed, and communication overhead are considered.

Secondly, **we propose new methodologies with the end-edge-cloud federated split learning framework**, where hardware-oriented optimal model splitting strategy, collaborative knowledge distillation mechanism, and hardware clustering-enabled semi-asynchronous federated learning approach are proposed.

Thirdly, **we achieve a novel implementation of edge intelligence**, where effective on-device load forecasting is carried out on a resource-constrained real-world hardware platform. To this end, we reckon our work can be seen as a pioneer in smart meter-based edge intelligence realization.

---

2. Will on-device intelligence for smart meters bring a lot of energy consumption?

Reply:

Thank you for raising the question about the energy consumption issue brought about by on-device intelligence in smart meters. Indeed, the computational demands of running machine learning algorithms and other advanced analytics may increase energy consumption. However, this slight increase in energy consumption is generally negligible compared to the benefits that on-device intelligence provides. This view can be supported by the following perspectives:

- The power consumption of smart meters is extremely low, as they mainly rely on simplified processor architecture and communication technology. Taking the STM32F407 microcontroller used in this paper as an example, when operating at a maximum frequency of 168MHz and a supply voltage of 3.3V, its dynamic power consumption is approximately 429-462mW, which is only one-thousandth that of a desktop computer. Moreover, smart meters can function in a low-power mode with a power consumption of 82.5mW. Assuming that each customer's smart meter operates at maximum power for an average of one hour per day, **the additional annual electricity consumption of the smart meter does not exceed 0.17kWh**.
- The smart meters with on-device intelligence transform collected data into knowledge, providing deeper insights into the past, and a better understanding of the future of energy usage. This enables consumers to make better-informed decisions regarding their energy consumption habits. Studies have shown that effective energy management can result in energy savings of up to 10-20% per year. For example, **a household that consumes an average of 10,000 kWh of electricity per year could potentially save between 1,000 and 2,000 kWh of electricity annually** through energy management. Such improvements are crucial for reducing carbon emissions and contributing to the global efforts in combating climate change.

Overall, while on-device intelligence for smart meters may require some energy to operate, the potential benefits in terms of energy management and consumption efficiency far outweigh the energy consumption of the device itself.

---

3. Only CNN and LSTM were tested as the benchmark, which weakens the significance of this work in terms of deep learning efficiency.

Reply:

Thanks for your comment on experimental backbones. We acknowledge your concern regarding using only CNN and LSTM as benchmarks for effectiveness validation. The results of the experiments with the MLP model as the network backbone have been reported in Table 1. Furthermore, we have **expanded our experiments to include both RNN and GRU models**, which are also commonly used in load forecasting. The visualization results can be found in Fig. 8. We have added the following discussion of the performance comparison for different methods on the several abovementioned network backbones to the revised manuscript.

We also implement our method and the benchmarks with several common deep learning-based backbones in load forecasting, including a convolutional neural network (CNN), recurrent neural network (RNN), gate recurrent unit (GRU), and long short-term memory (LSTM).

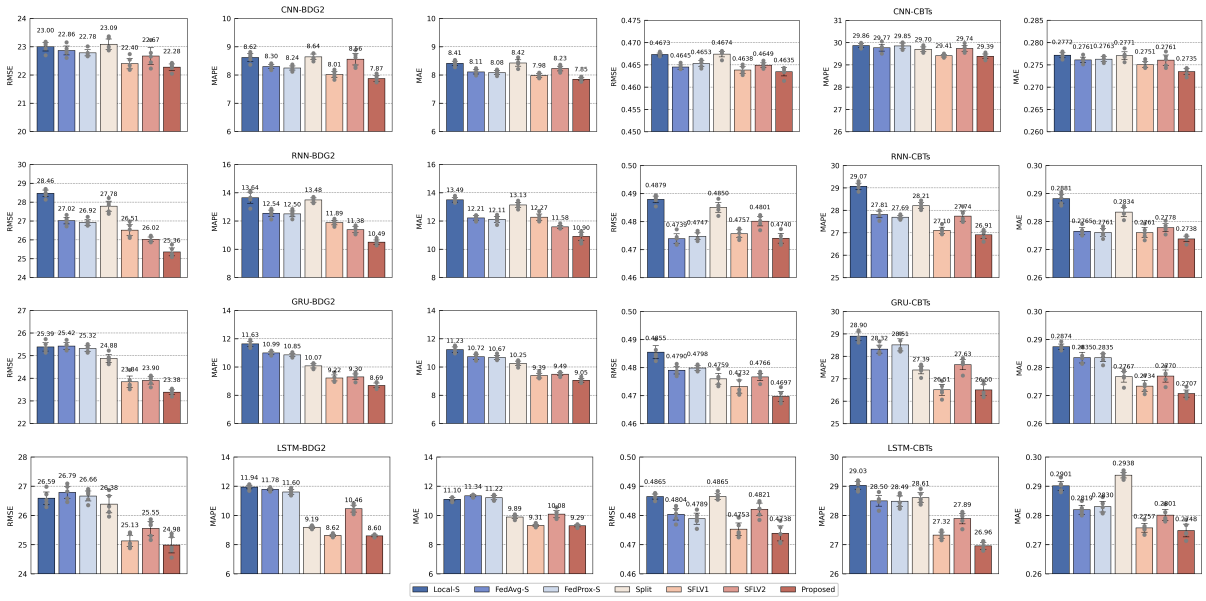


Fig. 8: Performance evaluation of the proposed method with different neural networks as the backbone. We compare the accuracy metrics of our method with other device-friendly methods with a CNN, RNN, GRU or LSTM as the backbone. The mean accuracy with 95% confidence intervals is presented with 5 independent experiments.

As shown in Fig. 13, our proposed method surpasses other feasible on-device methods on both BDG2 and CBTs datasets, which shows that our method is model-agnostic and performs well with different neural networks as the backbone. Recalling the results presented in Table 6, we see that the basic deep-learning model MLP even achieves higher forecasting accuracy. The possible reason is that MLP with simple architectures can accommodate more neurons than models with complex architecture in limited memory space, thus achieving a stronger representation capacity. In summary, we can conclude that the proposed method consistently maintains high performance in handling various real-world forecasting scenarios.

In addition, we have **added the improved federated learning method FedProx as a benchmark in our experiments** to illustrate the superiority of the proposed method. The results in Table 1 indicate that, beyond the same efficiency metrics, the improved FedProx and FedAvg algorithms do not differ significantly in terms of accuracy performance. This suggests that the overall data distribution among consumers is not significantly heterogeneous, so the proximal term introduced by FedProx has a limited effect on improving model performance. In summary, it can be concluded that the proposed method remains the best-performing of all the on-device feasible methods on both datasets.

Besides, we **conducted extensive experiments on all the methods considering a complete network with deeper 7 hidden layers**, where the results are reported in Supplementary Table 3. Comparing Tables 1 and Supplementary Table 3, we can observe that the employment of

Table 1: Performance of different methods on BDG2 and CBTs in terms of accuracy, memory, training time, and communication overhead per round.

Method	BDG2			CBTs			Memory (KB)	Training Time (s)	Communication (KB)
	RMSE	MAPE	MAE	RMSE	MAPE	MAE			
Cen	22.82	8.41	8.20	0.4780	26.98	0.2727	2578.42 (1.0×)	586.42 (2.05×)	-
Local-S	23.56	8.13	8.12	0.4698	28.19	0.2726	152.53 (16.9×)	41.19 (29.14×)	-
FedAvg-S	22.79	7.67	7.98	0.4681	27.04	0.2699	152.53 (16.9×)	46.49 (28.80×)	5.50 (27.18×)
FedProx-S	22.79	7.67	7.98	0.4681	27.04	0.2699	152.53 (16.9×)	46.49 (28.80×)	5.50 (27.18×)
Local-M	22.84	7.75	8.03	0.4645	26.67	0.2682	2578.42 (1.0×)	1187.15 (1.01×)	-
FedAvg-M	22.25	**6.96**	7.48	**0.4614**	25.81	0.2637	2578.42 (1.0×)	1200.44 (1.0×)	149.50 (1.0×)
FedProx-M	22.25	**6.96**	7.48	**0.4614**	25.81	0.2637	2578.42 (1.0×)	1200.44 (1.0×)	149.50 (1.0×)
Split	23.27	7.68	7.96	0.4679	26.83	0.2687	103.75 (24.8×)	96.00 (12.50×)	91.25 (1.63×)
SFLV1	22.34	7.25	7.68	0.4647	26.03	0.2664	103.75 (24.8×)	96.49 (12.44×)	96.75 (1.54×)
SFLV2	22.76	7.53	7.89	0.4674	26.49	0.2683	103.75 (24.8×)	96.49 (12.44×)	96.75 (1.54×)
Proposed	**22.17**	6.98	**7.44**	0.4630	**25.74**	**0.2636**	115.07 (22.4×)	62.41 (19.23×)	74.43 (2.01×)

Supplementary Table 3: Performance of different methods on BDG2 and CBTs in terms of accuracy, memory, training time, and communication overhead per round considering deeper neural network with 7 hidden layers.

Method	BDG2			CBTs			Memory (KB)	Training Time (s)	Communication (KB)
	RMSE	MAPE	MAE	RMSE	MAPE	MAE			
Local-M	22.75	7.68	8.03	0.4678	26.68	0.2721	5178.25 (1.0x)	4335.86 (1.01x)	-
FedAvg-M	22.33	**6.84**	7.44	**0.4617**	26.28	0.2638	5178.25 (1.0x)	4387.33 (1.0x)	761.5 (1.0x)
FedProx-M	22.42	7.01	7.52	0.4620	26.17	0.2641	5178.25 (1.0x)	4387.33 (1.0x)	761.5 (1.0x)
Split	22.93	7.89	8.14	0.4694	26.74	0.2713	103.75 (49.9x)	281.86 (15.55x)	91.25 (8.34x)
SFLV1	22.56	7.03	7.57	0.4963	26.43	0.2650	103.75 (49.9x)	282.84 (15.51x)	96.75 (7.87x)
SFLV2	22.75	7.18	7.66	0.4647	26.39	0.2663	103.75 (49.9x)	282.84 (15.51x)	96.75 (7.87x)
Proposed	**22.22**	6.86	**7.38**	0.4626	**26.12**	**0.2637**	115.07 (45x)	214.68 (20.43x)	74.43 (10.23x)

deeper networks has a rather limited or even negative gain in accuracy. In addition, for non-split methods like Local-M, FedAvg-M, and FedProx-M, deeper networks imply more memory footprint, training time and communication overhead for smart meters. Owing to the assistance of edge servers in the proposed framework, the burden on the smart meter does not increase with the depth of neural networks, as there are still only a few layers deployed on the smart meter, and the additional model layers are taken up by the edge servers. Remarkably, the results show that the proposed method also achieves the best performance among existing load forecasting methods when considering a deeper neural network with 7 hidden layers. Our framework effectively reduces peak memory by 45x, training time by 20.43x, and communication overhead by 10.23x.

We appreciate your insightful review and guidance, which has greatly contributed to the improvement of our work. By incorporating these additional case studies, we believe that our work now presents a more thorough evaluation of deep learning methods for load forecasting. We hope that the revised version of our study now addresses your concerns.

---

4. The two test datasets are all before 2018, which is not in a kind of up-to-date way.

Reply:

Thanks for the insightful comment on the test datasets. We understand your concerns regarding the recency of the two load datasets, BDG2 and CBTs. However, we would like to emphasize that for forecasting tasks the recency of the datasets is not the primary focus. The forecasting task focuses primarily on analyzing patterns and trends in the dataset, rather than just predicting a specific point in time. In this case, the recency of the dataset has less of an impact, as the general patterns and trends in the historical data are still of high reference value. Furthermore, these two datasets have become the benchmark in the field of consumer energy data analysis, akin to the role of the MNIST dataset in the computer vision domain. They have been extensively employed in numerous studies, demonstrating their significance and widespread application in the research community. We would like to clarify some points that highlight the comprehensiveness and relevance of these datasets for our study.

- **Data Duration and Data Quality**: Both BDG2 and CBTs datasets have been collected over an extensive period, ensuring the inclusion of diverse load patterns. The data quality is high as it has been meticulously recorded and verified by the respective organizations.
- **User Quantity and User Type**: These datasets cover a large number of consumers across different geographic locations and sectors, including residential, commercial, and industrial. This ensures that our analysis benefits from a wide variety of load profiles, enhancing the generalizability of our findings.

Supportive Table: Open-access electrical load datasets containing data collected after 2018.

Dataset	Year	Duration	Quality	Quantity	Type
Schlemminger et al. [1]	2018-2020	✓	✗	✓	✓
Lara et al. [2]	2023	✗	✓	✗	✓
Zhou al. [3]	2016-2021	✓	✗	✓	✗
Chavat et al. [4]	2019-2020	✓	✗	✓	✓
CEUS [5]	2018-2022	✓	✓	✓	✗
CoSSMic [6]	2014-2019	✓	✗	✓	✓
JERICHO-E-usage [7]	2019	✗	✓	✓	✓
IDEAL [8]	2016-2018	✓	✗	✓	✓
ELMAS [9]	2018	✗	✓	✓	✓

Despite being collected before 2018, these two widely used datasets provide valuable insights

into load patterns and are considered reliable sources for analysis. We acknowledge that more recent datasets could potentially offer additional perspectives. The supportive table reviewed some open-access electrical load datasets collected after 2018. We found that these datasets may have some limitations and cannot be applied to our study. For example, the latest dataset of 38 residential loads published in [1] has data availability of less than 90% for 15 users, potentially affecting the reliability of the analysis. In addition, the number of consumers [2] or the duration [7, 9] contained in some datasets can not support the need for 1.5-year data from 30 smart meters in this paper. Dataset [3] contains data for commercial consumers only while dataset [5] includes data for industrial consumers only, which limits the applicability of the findings to specific sectors or regions.

In conclusion, we believe that the comprehensiveness of BDG2 and CBTs datasets, in terms of duration, data quality, user quantity, and user types, outweighs the potential benefits of using more recent, but possibly less representative datasets. We will continue to monitor new datasets and consider incorporating them into our research as they become available and meet our quality criteria. Thank you once again for your valuable input, and we hope that this explanation addresses your concern.

[1] Schlemminger M, Ohrdes T, Schneider E, et al. Dataset on electrical single-family house and heat pump load profiles in Germany[J]. Scientific data, 2022, 9(1): 56.

[2] Lara E G, Díaz A V, Mariñez C N P. Electrical dataset of household appliances in operation in an apartment[J]. Data in Brief, 2023, 51: 109742.

[3] Zhou K, Hu D, Hu R, et al. High-resolution electric power load data of an industrial park with multiple types of buildings in China[J]. Scientific Data, 2023, 10(1): 870.

[4] Chavat J, Nesmachnow S, Graneri J, et al. ECD-UY, detailed household electricity consumption dataset of Uruguay[J]. Scientific Data, 2022, 9(1): 21.

[5] Commission, C. E. California Commercial End-Use Survey. https://www.energy.ca.gov/data-reports/surveys/californiacommercial-end-use-survey

[6] Data, O. P. S. Data Platform – Open Power System Data. https://data.open-power-system-data.org/household_data/

[7] Priesmann J, Nolting L, Kockel C, et al. Time series of useful energy consumption patterns for energy system modeling[J]. Scientific Data, 2021, 8(1): 148.

[8] Pullinger M, Kilgour J, Goddard N, et al. The IDEAL household energy dataset, electricity, gas, contextual sensor data and survey data for 255 UK homes[J]. Scientific Data, 2021, 8(1): 146.

[9] Bellinguer K, Girard R, Bocquet A, et al. ELMAS: a one-year dataset of hourly electrical load profiles from 424 French industrial and tertiary sectors[J]. Scientific Data, 2023, 10(1): 686.

---

5. Although the hardware platform was provided, it's more like a demo with a very limited scale. It would be more meaningful if the technique could step out of the lab.

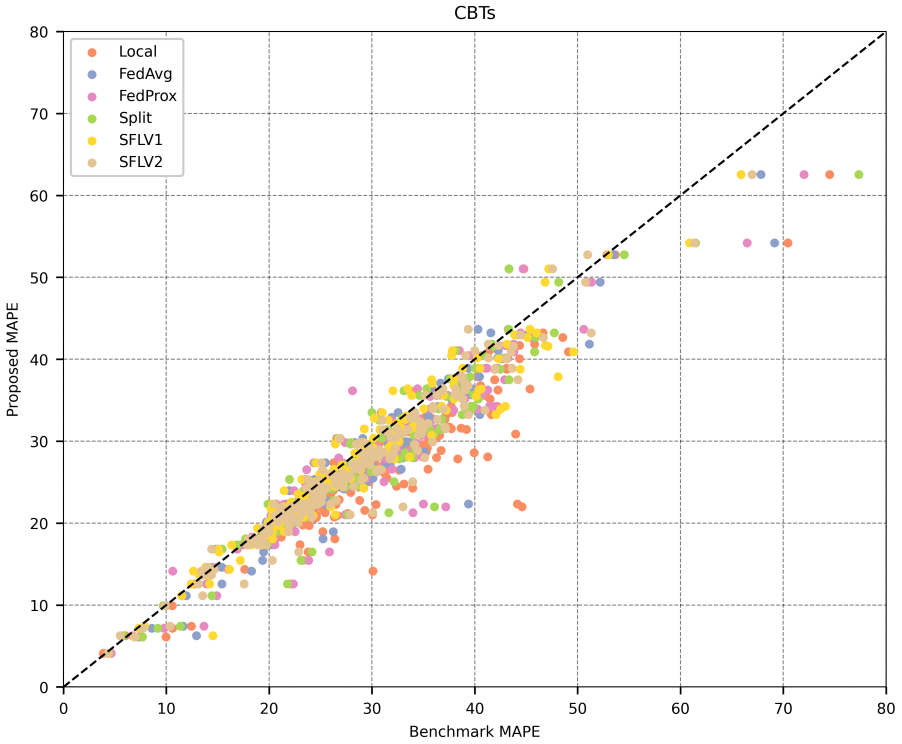
Reply:

Thank you for the valuable comment regarding the limited scale of our initial experiment. We have already expanded the scope of our experiment by conducting large-scale tests using 200 smart meters, which represent approximately the size of an energy community. By studying their energy consumption patterns, we can gain insights into the effectiveness and applicability of our approach in real-world settings. The performance comparison of the benchmark methods and the proposed method on 200 individual households can be visualized in Supplementary Fig. 10, where the x-axis and the y-axis represent the performance of each method in terms of MAPE respectively. The dashed line in the figure denotes that both models perform equally well. Households on which the proposed model performs better than the benchmark models are indicated by the points below the dashed line. The fact that most of the points are below the dashed line, with some of them much below, implies that the proposed model can perform well on the majority of households and achieve significant improvement on a few of them.

Regarding the transition from the laboratory to the industry, we would like to emphasize that at this stage this research focuses on theoretical innovations. We acknowledge that the direct application of our method to the grid may face several challenges, including compliance with national laws and regulations, consumer data security and privacy, and system integration and accessibility, which is beyond the scope of this paper. To tackle these issues, we have already engaged with smart meter manufacturers, who have expressed strong interest in our research after a thorough discussion. Additionally, we have reached out to the relevant government departments to introduce our preliminary research ideas and achievements, aiming to facilitate the establishment of industry standards and ensure that our approach aligns with the legal and regulatory requirements. We hope to showcase our achievements, allowing more people to see and adopt our techniques. Furthermore, we envision providing a reference for the development of future smart meter intelligence applications, providing a basis for subsequent research and development in this field.

---

6. There is not a strong connection between the smart meter and renewable energy. It's better to make it clear at what extent/how large the improvement is of smart meter-supported demand flexibility for the RES promotion.



Supplementary Fig.10: Accuracy comparison of benchmark methods and proposed method in a large-scale scenario. The performance of the benchmark models is indicated by the x-axis, and the y-axis indicates that of the proposed model. Points below the dashed line indicate households where the proposed model performs better than the baseline model. The fact that most of the points are below the dashed line implies that the proposed model can perform well in the majority of households and achieve significant improvement in a few of them.

Reply:

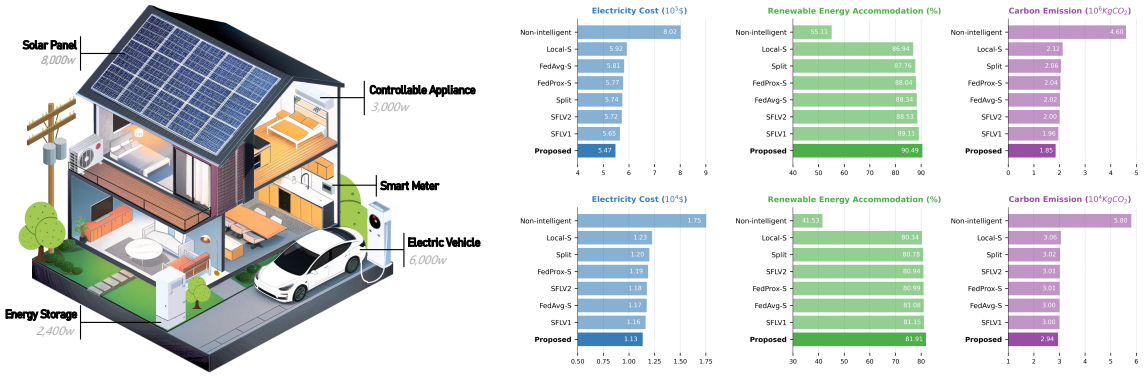
Thanks for your comment and valuable suggestion. Smart meters are the main driver of renewable energy accommodation by utilizing demand-side flexibility [1], which is achieved from two perspectives. First, the power system operator needs to carry out a demand-response program by estimating demand response potential [4] and designing dynamic price [5] to guide consumers to adapt their consumption behaviors in response to the volatile renewable energy generation, such as encouraging users to shift energy-intensive tasks to times when renewable generation is high. Smart meters are then the key element to the success of demand-response by providing data for analyzing consumers' behaviors and characteristics. Second, smart meters have the built-in ability to disconnect and reconnect particular loads remotely, and they can be used to regulate users' devices and appliances to manage demands and loads within "smart homes" in the future. To this end, smart meters can act as agents for home energy management systems to monitor the distributed renewable energy generation, storage, and consumption [6]. In conclusion, we reckon there is a strong connection between the smart me-

ter and renewable energy and this paper is trying to achieve intelligence based on smart meter hardware to facilitate utilizing demand-side flexibility for renewable energy accommodation.

Harnessing demand-side flexibility is a cost-effective strategy for promoting renewable energy accommodation [1], where smart meters play a pivotal role in this process. Smart meters are the core part of the advanced metering infrastructure in power systems, which are supported by sensors, control devices, and dedicated communication infrastructure [2]. Smart meters can record real-time energy information, including voltage, frequency, and energy consumption, from the demand side and can enable bidirectional communication between system operators and end-users [3]. The advanced functions of smart meters provide a strong foundation for harnessing demand-side flexibility in terms of data and hardware platforms [4,5]. On the one hand, smart meter data enable the estimation of demand response potential [6] and dynamic pricing design [7] to integrate renewable energy. On the other hand, smart meters can act as agents for home energy management systems to monitor the distributed renewable energy generation, storage, and consumption [8].

To quantify the impact of edge intelligence on downstream decision-making, we further investigated the impact of the proposed on-device forecasting approach on individual household energy management and demonstrated its great effectiveness in **reducing electricity cost (31.79%), promoting renewable energy accommodation (35.38%), and lowering carbon emissions (59.78%).** The following discussions for experimental results have been added to the revised manuscript.

Load forecasting facilitates consumers to gain deeper insights into future energy consumption patterns, thereby supporting tailored energy management decisions. In addition to conducting accuracy analysis, we explore the impact of forecasting errors on the downstream decision-making process. Fig. 9(a) illustrates a representative home, which features distributed flexible energy resources, including solar panels, controllable appliances, electric vehicles, and energy storage systems. Note that a building can be regarded as a multi-household collective with larger-scale distributed resources. Building energy management (BEM) and home energy management (HEM) are typically achieved through two stages: 1) short-term scheduling and 2) real-time balancing. Briefly, short-term scheduling aims to minimize electricity costs while ensuring a balance between forecast demand and supply by scheduling various flexible resources for upcoming periods. To this end, the smart meter installed on each building/home first predicts the future load using a pre-trained forecasting model and retrieves the time-of-use tariff information from the grid operator's cloud platform. On this basis, the smart meter can determine the operating strategies of storage systems and household appliances and recommend strategies for participating in the energy market. To save electricity costs, storage systems and electric vehicles can be charged during off-peak tariff periods, while grid-connected electricity sales



(a) (b)

Fig. 9: Impacts of different forecasting methods on individual energy management. (a) Schematic diagram of edge home energy management for flexible energy resources. Edge intelligence enables smart meters to manage local energy storage, controllable household appliances, electric vehicle charging, and energy market participation based on predicted loads. (b) Comparison of the electricity cost, renewable energy accommodation ratio, and carbon emission for a non-intelligent strategy and various edge intelligent methods. The experiments were conducted on 30 buildings and houses in the BDG2 and CBTs datasets for 180 test days.

can be conducted during peak solar generation periods. However, due to prediction errors, smart meters may require further adjustments to achieve the real-time supply and demand balance. In this case, a higher predicted load implies that consumers discard unused generated renewable energy, while a lower predicted load means that consumers have to temporarily purchase electricity in the energy market. Both situations are unfavourable for efficient energy management. In short, accurate forecasting results contribute to low additional grid electricity purchases and a high accommodation ratio of solar generation, thus reducing total carbon emissions.

We conducted comprehensive experiments on the BDG2 and CBTs datasets to showcase the effectiveness of the proposed method in enhancing decision-making for BEM and HEM. Fig. 9(b) provides a performance comparison of a non-intelligent strategy and various edge intelligent methods in terms of the electricity cost, renewable energy accommodation ratio, and carbon emission. In the non-intelligent strategy, smart meters without edge intelligence cannot provide any assistance or support for customers to schedule flexible energy resources. The results clearly show that introducing edge intelligence to smart meters can, on average, reduce electricity cost by 31.79%, increase renewable energy accommodation by 35.38%, and reduce carbon emission by 59.78% for each building. These improvements brought to each house can be found at 35.42%, 40.38%, and 49.31%, respectively. By adopting our approach, electricity cost savings of $1,176.11 per building and electricity cost savings of $18.93 per household can

be expected annually. Importantly, the proposed method, which has the highest forecasting accuracy among all intelligent methods, also achieves a significant performance improvement in individual energy management. Compared to the best-performing benchmarks, our approach saves electricity cost, boosts renewable energy consumption, and reduces carbon emission for buildings and houses, reaching values of 3.08%, 1.38%, 5.42% and 2.41%, 0.76%, 1.96%, respectively. Interestingly, the prediction error is not strictly monotone with the downstream decision cost. For instance, SFLV2 outperforms FedAvg-S in residential load forecasting, but its performance in subsequent energy management is unsatisfactory.

[1] O'Shaughnessy E, Shah M, Parra D, et al. The demand-side resource opportunity for deep grid decarbonization[J]. Joule, 2022, 6(5): 972-983.

[2] Avancini D B, Rodrigues J J P C, Martins S G B, et al. Energy meters evolution in smart grids: A review[J]. Journal of cleaner production, 2019, 217: 702-715.

[3] Mohassel R R, Fung A, Mohammadi F, et al. A survey on advanced metering infrastructure[J]. International Journal of Electrical Power & Energy Systems, 2014, 63: 473-484.

[4] Barai G R, Krishnan S, Venkatesh B. Smart metering and functionalities of smart meters in smart grid-a review[C] 2015 IEEE Electrical Power and Energy Conference (EPEC). IEEE, 2015: 138-145.

[5] Wang Y, Chen Q, Hong T, et al. Review of smart meter data analytics: Applications, methodologies, and challenges[J]. IEEE Transactions on Smart Grid, 2018, 10(3): 3125-3148.

[6] Dyson M E H, Borgeson S D, Tabone M D, et al. Using smart meter data to estimate demand response potential, with application to solar energy integration[J]. Energy Policy, 2014, 73: 607-619.

[7] Cai Q, Xu Q, Qing J, et al. Promoting wind and photovoltaics renewable energy integration through demand response: Dynamic pricing mechanism design and economic analysis for smart residential communities[J]. Energy, 2022, 261: 125293.

[8] Zhou B, Li W, Chan K W, et al. Smart home energy management systems: Concept, configurations, and scheduling strategies[J]. Renewable and Sustainable Energy Reviews, 2016, 61: 30-40.

Thanks again to all the reviewers for the valuable suggestions and comments. They helped us to improve the quality of the paper.

Reviewer 1 (Remarks to the Author):

The authors' implementation of a hardware testbed for edge intelligent smart meters is notable and is a good contribution to the field. I feel satisfied with the authors' response to my original comments. Their alterations are comprehensive and significant.

Reply:

Thank you for the insightful comments and positive feedback on our contribution to smart meter intelligence.

Reviewer 1 (Remarks on code availability):

The code is well-documented and provides enough information and instructions to reproduce the authors' results.

Reply:

Thank you for your positive feedback on the reproducibility and clarity of our codes.

Reviewer 2 (Remarks to the Author):

I appreciate the efforts spent by the authors on revising this paper and addressing my questions.

Compared to the previous version, the quality of this manuscript is improved. However, there are still several points to be considered:

Reply:

Thank you for acknowledging the improvements made in our manuscript and for providing additional feedback. We appreciate the time and effort you have taken to review our work. We will now respond point-by-point to your comments and questions in the following.

---

1. The limitations of the communication and computational capabilities of edge devices are well-known challenges in the edge environment. Thus, there are a number of existing studies focusing on communication and computational resource optimization, especially papers published in IEEE Transactions, and many international conferences. Thus, the statement "previous studies have often overlooked the limitations of the communication and computational capabilities of edge devices" in the last paragraph on page 3 is not proper. I agree that the implementation of smart meters would be harder than other edge devices, however, the authors need to explicitly demonstrate the special challenge their approach tackled.

Reply:

Thank you for pointing out the potential issues regarding the clarification of the unique challenges our approach addresses for smart meter intelligence. We acknowledge the existence of numerous studies focusing on communication and computational resource optimization in edge environments. However, in this paper we aim to address the specific challenges that arise when deploying edge intelligence on smart meters.

Existing research on edge intelligence that considers the communication and computational capacities of edge devices can be roughly categorized into two areas. 1) Focusing on model compression techniques, including pruning [1], quantization [2], knowledge distillation [3], low-rank decomposition [4], Huffman coding [5], and so on. Typically, this approach shrinks the size of the compact model, reducing the computational demand on edge devices, but it also comes with a sacrifice in model accuracy. 2) Splitting large models into multiple smaller sub-models through split learning [6], which can be distributed between edge devices and the cloud for collaborative computation. This approach aims to reduce the computational burden on individual devices while maintaining model performance. While these approaches may be effective in some scenarios, further research and optimization may be needed in the application of smart

meter edge intelligence.

Introducing edge intelligence to smart meters has three characteristics. 1) Large-scale device heterogeneity: the computational capacities and communication conditions of ubiquitous smart meters vary widely due to task occupancy, physical connection management, and device installation time. This heterogeneous characteristic poses challenges to the synchronous computation of smart meter edge intelligence. 2) Electricity consumption behavior similarity: The distribution of load data collected by smart meters from different customers may be relatively close due to their similar electricity consumption behaviors and patterns. This similarity in consumption behavior brings opportunities for the mutual gain of multiple consumers' smart meter data. 3) AMI system edge-end architecture: existing advanced metering infrastructure (AMI) integrates a large number of edge devices and establishes a two-way communication network between smart meters and edge nodes. This IoT architecture lays the foundation for collaboration between the edge servers and the end smart meters.

Motivated by the abovementioned characteristics, we propose a unified, comprehensive federated split learning framework incorporating federated learning and split learning **tailored for smart meter intelligence**. The proposed method facilitates ubiquitous smart meters to achieve edge intelligence by collaboratively utilizing distributed data with the assistance of edge servers. **In particular, we address the challenges of computation offloading, device collaboration, and heterogeneous aggregation in our framework**. Specifically, our optimal splitting ratio explores how to efficiently split the entire model while ensuring that smart meter memory overflow is avoided. Moreover, our collaborative training approach incorporates a knowledge distillation mechanism that enables smart meters to train models in parallel across different entities. Finally, our heterogeneous aggregation method addresses the delay issues caused by varying computation power in large-scale heterogeneous smart meters during model aggregation.

We have revised the specific challenges our approach addresses in the manuscript as follows:

While previous edge intelligence studies cannot be applied to the smart grid since the ubiquitous smart meters present unique challenges and opportunities. Our work provides a comprehensive solution tailored for smart meter hardware that translates theoretical methods into practical, real-world applications. This paper focuses on two critical questions in achieving on-device intelligence: "How can we efficiently utilize distributed data?" and "How can we train models on resource-constrained devices?". To answer these questions, we present an end-edge-cloud framework that combines federated learning and split learning to intellectualize resource-constrained smart meters for on-device load forecasting in a privacy-enhancing manner. This work overcomes the constraints inherent to smart meter environments, ensuring that our approaches are not only theoretically sound but also viable for on-the-ground deploy-

ment. In particular, we develop an optimal splitting strategy, collaborative knowledge distillation mechanism, and semi-asynchronous aggregation approach in our framework to tackle the issues of computation offloading, device collaboration, and heterogeneous aggregation for smart meter intelligence.

[1] Jiang Y, Wang S, Valls V, et al. Model pruning enables efficient federated learning on edge devices[J]. IEEE Transactions on Neural Networks and Learning Systems, 2022, 34(12): 10374-10386.

[2] Chen W, Qiu H, Zhuang J, et al. Quantization of deep neural networks for accurate edge computing[J]. ACM Journal on Emerging Technologies in Computing Systems (JETC), 2021, 17(4): 1-11.

[3] Hao Z, Luo Y, Wang Z, et al. Model compression via collaborative data-free knowledge distillation for edge intelligence[C]//2021 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2021: 1-6.

[4] Shi Y, Zhang J, Chen W, et al. Generalized sparse and low-rank optimization for ultra-dense networks[J]. IEEE Communications Magazine, 2018, 56(6): 42-48.

[5] Han S, Mao H, Dally W J. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding[J]. arXiv preprint arXiv:1510.00149, 2015.

[6] Poirot M G, Vepakomma P, Chang K, et al. Split learning for collaborative deep learning in healthcare[J]. arXiv preprint arXiv:1912.12115, 2019.

---

2. In addition, the contents in Fig. 1 are not exactly the same as the statements in the paper. The authors declare the privacy advantage achieved by federated learning, but this is not mentioned in the overview. In addition, the overlapping part is supposed to be the common characteristics or advantages of federated learning and split learning. However, this figure should clearly show how federated learning and split learning can enhance the performance of each other in various dimensions. The current version needs to be modified.

Reply:

Thank you for pointing out the discrepancies between Fig. 1 and the statements in the manuscript. We understand the need to better represent the common characteristics of federated learning and split learning and how their combination can enhance performance in various dimensions.

We have revised Fig. 1 to compare mainstream energy analysis methods in terms of accuracy, memory, communication, computation, and privacy. The method characteristics are rated as excellent, good, fair, or poor for each method. We can observe that the traditional centralized learning approach requires raw sensitive data collected by smart meters, causing consumer pri-

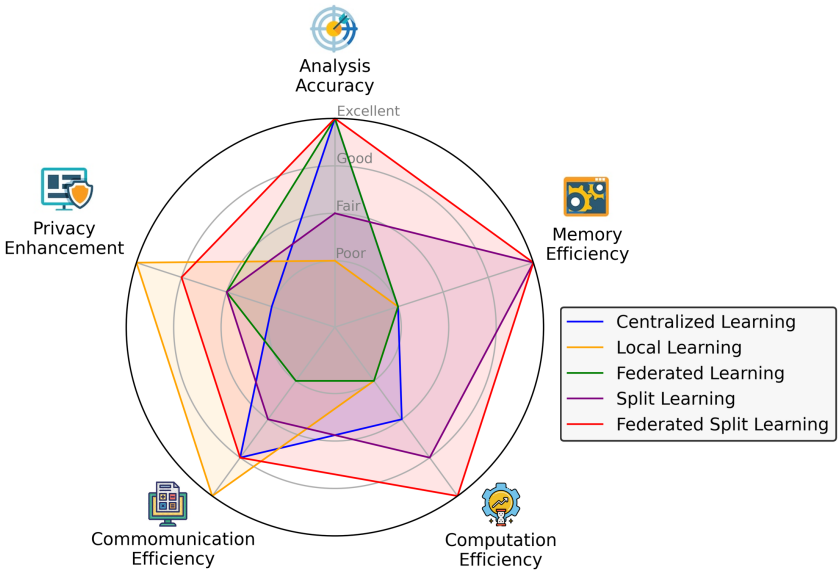


Fig. 1: Comparison among mainstream energy data analytics methods. The excellent, good, fair, and poor coordinate points represent the general performance of different methods in various dimensions. Our proposed federated split learning integrates the advantages of federated learning and split learning methods, achieving exceptional performance across all dimensions.

vacy leakage. The advantage of the local learning approach is that smart meters do not require any information interaction with other entities, thereby eliminating communication overhead and privacy leakage. However, such methods generally achieve unsatisfactory performance in analysis accuracy due to insufficient data resources and constrained hardware resources. By contrast, the federated learning approach can utilize data stored by multiple devices for model training in a privacy-enhancing way to improve analysis accuracy, and the split learning approach can migrate most of the model training burden to high-capacity servers without raw data sharing to reduce the on-device memory footprint and computation time. **Notably, our proposed federated split learning integrates the advantages of federated learning and split learning methods, achieving exceptional performance across all dimensions.**

we have added the following description for Fig. 1 in the revised manuscript:

Fig. 1 compares the characteristics of mainstream learning methods, highlighting that our framework consolidates several properties: higher accuracy, reduced memory footprint, faster computation speed, smaller communication overhead, and enhanced privacy.

---

3. Another point is about privacy-enhancing performance. The metrics about privacy are not clear in the experiments. Generally, privacy performance can be measured in multiple ways, such as whether the desired data can be figured out by privacy attacks like membership inference attacks, etc. As the authors claim that this is part of their contributions, it would be

necessary to explicitly demonstrate the results to the readers.

Reply:

Thank you for your insightful comments and for raising concerns about privacy-enhancing performance. We understand the importance of providing clear and explicit metrics for evaluating the privacy performance of our proposed method.

Indeed, both federated learning and split learning require uploading model information to the server to seek participation and assistance from other entities. Honest and curious servers may attack users' original data based on this sensitive information, leading to privacy leakage. Extensive research has designed specific attack models for federated learning and split learning [1-4]. Only a few studies have attempted to explore attack methods in emerging federated split learning [5]. Due to the mismatch in the number of studies, we cannot compare the privacy performance of these methods through qualitative analysis.

As the reviewer pointed out, the membership inference attack [6] is a common black-box attack method in decentralized learning, which aims to determine whether a data point was part of the training set of a machine learning model by analyzing the weight parameters or hidden layer activations. **For a fair comparison, we conduct experiments to analyze the privacy leakage degree of different distributed methods under membership inference attacks.** Specifically, the attacker would construct a shadow model to mimic the behaviour of the target model. Then the attacker creates a binary classification model called the attack model to distinguish between members and non-members. For instance, the attacker can use the gradients for the samples (for federated learning) or intermediate activations (for split learning) of the target model as input features for the attack model. Finally, the attacker uses the trained attack model to infer the membership status of the data points of each electricity consumer. We can evaluate the performance of the attack model on different methods using standard classification metrics, i.e. accuracy.

Supplementary Table 4 Performance evaluation of different methods on BDG2 and CBTs under passive membership inference attack.

Method	Attack Accuracy	
	BDG2	CBTs
FedAvg-S	0.6738	0.4804
Split	0.8387	0.4628
SFLV1	0.5783	0.3894
SFLV2	0.6012	0.4152
Proposed	**0.5320**	**0.3849**

The comparison of attack accuracy on the two datasets is shown in Supplementary Table 4. We can observe that the proposed method demonstrates significant enhancement against privacy attacks compared to the benchmarks. The possible reason is that the gradients of the feature extractor and the feature processor are computed independently without direct correlation in the parallelization mechanism of our approach. The attacker cannot accurately calculate the gradient value for the samples during the training process. **We can conclude the proposed method yields privacy-enhancing performance on the two datasets.**

We would like to emphasize that the time-domain correlation of load data is crucial as it represents the time-series information. Therefore, even if the attackers determine the membership of load data for a particular customer through the attack models, they cannot further infer the customer's electricity usage pattern. Compared to uploading raw data in centralized learning, the proposed method facilitates significant enhancement of consumer privacy information.

[1] Nasr M, Shokri R, Houmansadr A. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning[C]//2019 IEEE symposium on security and privacy (SP). IEEE, 2019: 739-753.

[2] Lyu L, Yu H, Yang Q. Threats to federated learning: A survey[J]. arXiv preprint arXiv:2003.02133, 2020.

[3] Pasquini D, Ateniese G, Bernaschi M. Unleashing the tiger: Inference attacks on split learning[C]//Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. 2021: 2113-2129.

[4] Liu J, Lyu X, Cui Q, et al. Similarity-based label inference attack against training and inference of split learning[J]. IEEE Transactions on Information Forensics and Security, 2024.

[5] Zhang Z, Pinto A, Turina V, et al. Privacy and efficiency of communications in federated split learning[J]. IEEE Transactions on Big Data, 2023, 9(5): 1380-1391.

[6] Shokri R, Stronati M, Song C, et al. Membership inference attacks against machine learning models[C]//2017 IEEE symposium on security and privacy (SP). IEEE, 2017: 3-18.

---

4. The last point is about the limitations of their approach. The authors provide their discussions in Section 3, which is good. However, the proposed approach has clearly limitations in the implementations and also the techniques adopted. I understand that it would be hard to have a perfect design, however, the discussions about the approach limitations are also a significant contribution and key idea to be delivered to the readers. Thus, I would recommend the authors provide such discussions to enhance the quality of this paper.

Reply:

Thanks for your constructive comment on the limitations of the proposed approach. We agree that discussing the limitations will provide a more comprehensive and balanced view of our work. Despite the end-edge-cloud framework bringing advancements in smart meter intelligence, this study has two concerns that should be addressed in industrial applications. 1) The adopted clustering algorithm is designed for the computing resources of heterogeneous devices without considering the geographical location and physical connection characteristics of smart meters in smart grids. 2) The established experimental platform only selects one representative hardware configuration for the smart meter. Implementing smart meter intelligence should consider the compatibility of devices equipped with varying core models and communication technologies.

We have added the following limitation discussion on the technique design and platform implementation in the manuscript:

Despite the end-edge-cloud framework bringing advancements in smart meter intelligence, this study has two concerns that should be addressed in industrial applications. First, the adopted clustering algorithm is designed for the computing resources of heterogeneous devices without considering the geographical location and physical connection characteristics of smart meters in smart grids. Second, the established experimental platform only selects one representative hardware configuration for the smart meter. Implementing smart meter intelligence should consider the compatibility of devices equipped with varying core models and communication technologies.

Reviewer 3 (Remarks to the Author):

The authors provide good responses to my concerns. I have some further but relatively minor comments in this turn.

Reply:

Thanks for your thoughtful comments and constructive suggestions on our manuscript. We appreciate the time and effort you have taken to review our work. We will now respond point-by-point to your comments and questions.

---

1. Even though I could see there is a relatively light energy requirement for the smart meters, the comparison of power consumption between the smart meter and household energy is kind of not at the same level. Smart meters + cloud server + edge serve VS total house energy saving will be a better comparison. In this case, is there a possibility that the consumed energy by the whole system (cloud + edge servers + meter) is larger than the saved energy? I believe there is a need for the scale balance here.

Reply:

Thank you for your insightful comments and suggestions. We understand your concerns about the comparison of power consumption between the energy consumed by the whole system and the total house energy saving. To address your questions, we conduct a comprehensive energy consumption analysis as follows:

- **Energy consumption of the edge intelligence system:** The source of energy consumption of the whole system can be categorized into training and inference phases. The cloud server, edge servers, and smart meters need to collaborate for 100 rounds of model training to update parameters at a certain interval (assumed to be one month). Since the whole model will be deployed to the smart meter after completing the model training, the inference phase is done by the smart meter only. The device energy consumption can be calculated by multiplying the operating time by the operating power. Take our platform, which consists of 30 smart meters, 10 PCs, and 1 tower server, as an example. The dynamic power of the STM32F405RGT6 microcontroller is 429-462mW [1]. The edge servers and cloud server are rated to operate at approximately 180W and 800W, respectively. The computational and communication times per training round for the edge server and smart meter are 60.16 seconds and 58.29 seconds, respectively. Since the computation of additive operations in model aggregation is negligible for cloud servers, we only consider the communication time of cloud servers, which is less than 1 second measured in our experiments. The additional annual electricity consumption of the pro-

posed system for model training can be calculated as the sum of that of the cloud server, 3 edge servers, and 30 smart meters, i.e., $(800\text{W} \times 1\text{s} + 3 \times 180\text{W} \times 60.16\text{s} + 30 \times 0.5\text{W} \times 58.29\text{s}) \times 100\text{rounds} \times 12\text{months} = 11.387\text{kWh}$. Besides, the smart meter spends 5.3 seconds per hour performing model inference for load forecasting. The additional annual electricity consumption of the proposed system for model inference can be calculated as $30 \times 0.5\text{W} \times 5.3\text{s} \times 24\text{hours} \times 365\text{days} = 0.193\text{kWh}$. **Hence, the total energy consumption of the whole system is estimated to be 11.58kWh.**

- **Energy consumption saving of the homes:** The smart meters with on-device intelligence transform collected data into knowledge, providing deeper insights into the past, and a better understanding of the future of energy usage. This enables consumers to make better-informed decisions regarding their energy consumption habits. In our experiments, the average daily PV generation of the household solar panel in the CBTs dataset is 23.25kWh. The total annual power generation of the selected 30 households can be calculated as $30 \times 23.25\text{kWh} \times 365\text{days} = 254587.5\text{kWh}$. The results in our experiments have shown that effective energy management can improve renewable energy accommodation in a household by 40.38%. **Hence, the smart meter intelligence potentially saves 102802.432kWh of electricity annually through home energy management.**

Overall, while edge intelligence for smart meters may require some energy to operate, the potential benefits in terms of energy management and consumption efficiency far outweigh the device energy consumption of the cloud server, edge servers, and smart meters.

[1] https://www.st.com/resource/en/datasheet/stm32f405rg.pdf.

---

2. IMHO, it's more convincing to compare the energy consumption between the conventional way (e.g. centralized computation/learning method) and edge intelligence FL-based smart meter method if you wanna show the reduced energy consumption. Do you have any technical support for the assumption of '1/24 maximum power operation of the smart meters'?

Reply:

Thanks for your comment on the energy consumption comparison between smart meter intelligence and traditional methods. We would like to clarify that our primary focus is not to emphasize the energy reduction aspect. The edge intelligence approach mainly relies on low-power smart meters and edge servers for computation, while centralized methods perform computations on energy-intensive cloud servers. The computational and communication times per training round for the edge server and smart meter are 60.16 seconds and 58.29 seconds, respectively. Due to the benefits of distributed device parallel computation, the edge intelligence method has a faster training time than centralized methods (586.42 seconds). The energy con-

sumption required for training once with our proposed method is $(800\text{W} \times 1\text{s} + 3 \times 180\text{W} \times 60.16\text{s} + 30 \times 0.5\text{W} \times 58.29\text{s}) \times 100\text{rounds} = 0.965\text{kWh}$, while the centralized method consumes $800\text{W} \times 586.42\text{s} \times 100\text{rounds} = 13.03\text{kWh}$. **In brief, the proposed method achieves better performance in energy consumption compared to traditional centralized method.**

In practical scenarios, it is unrealistic for smart meters to solely run edge intelligence algorithms for a while as they have to perform other functions such as energy metering. Based on engineering experience with task occupancy, we set the operational frequency from 1/2 to 1/8 of the maximum values.

We have added the following description for the device frequency setting in the revised supplementary information:

The operational frequency settings of smart meters in our experiments range from 1/2 to 1/8 of the maximum values, which is an engineering experience value derived from the perspective of task occupancy rate.

---

3. How did you calculate the reduced electricity cost, increased renewable energy accommodation, and reduced carbon emission?

Reply:

Thanks for your comment on the improvement calculation. Building energy management (BEM) and home energy management (HEM) are typically achieved through two stages: 1) short-term scheduling and 2) real-time balancing. Briefly, short-term scheduling aims to minimize electricity costs while ensuring a balance between forecast demand and supply by scheduling various flexible resources for upcoming periods. To this end, the smart meter installed on each building/home first predicts the future load using a pre-trained forecasting model and retrieves the time-of-use tariff information from the grid operator's cloud platform. On this basis, the smart meter can determine the operating strategies of storage systems and household appliances and recommend strategies for participating in the energy market.

The objective of the short-term scheduling is to schedule the energy consumption of home appliances to help consumers reduce electricity costs based on forecasted load, which can be expressed as:

$$\min C = \sum_{t=1}^{\mathrm{T}} \lambda_{\mathrm{t}} \left(P_{\mathrm{t}}^{\mathrm{fcst}} + P_{\mathrm{t}}^{\mathrm{cont}} + P_{\mathrm{t}}^{\mathrm{EV}} + P_{\mathrm{t}}^{\mathrm{ESS}} - P_{\mathrm{t}}^{\mathrm{solar}} \right) \tag{1}$$

where T denotes the scheduling time scale; λ_t denotes the time-of-use electricity price at time t; P_t^{fcst} denotes the forecasted load consumption at time t; $P_{\mathrm{t}}^{\mathrm{AC}}$, $P_{\mathrm{t}}^{\mathrm{EV}}$, $P_{\mathrm{t}}^{\mathrm{ESS}}$, and $P_{\mathrm{t}}^{\mathrm{solar}}$ denote the power of air conditioner (AC), electric vehicle (EV), energy storage system (ESS), and solar panel at time t, respectively. The positive and negative signs of $P_{\mathrm{t}}^{\mathrm{ESS}}$ correspond to the discharge and

charging states of the ESS.

The feasibility constraints limit the operating power of appliances within a feasible range, which can be formulated as follows:

$$\begin{aligned} P_{\text{min}}^{\text{AC}} &\leq P_{\text{t}}^{\text{AC}} \leq P_{\text{max}}^{\text{AC}} \\ P_{\text{min}}^{\text{EV}} &\leq P_{\text{t}}^{\text{EV}} \leq P_{\text{max}}^{\text{EV}} \\ -P_{\text{max}}^{\text{ESS}} &\leq P_{\text{t}}^{\text{ESS}} \leq P_{\text{max}}^{\text{ESS}} \end{aligned} \tag{2}$$

where $P_{\text{min}}^{\text{AC}}$ and $P_{\text{max}}^{\text{AC}}$ denote the minimum and maximum operating power of the AC, respectively; $P_{\text{min}}^{\text{EV}}$ and $P_{\text{max}}^{\text{EV}}$ denote the minimum and maximum charging power of the EV, respectively; $P_{\text{max}}^{\text{ESS}}$ denote the maximum charging and discharging power of the ESS.

The thermal dynamics constraints restrict the indoor temperature within a comfortable range, which can be formulated as follows:

$$\begin{aligned} T_{t+1}^{\text{in}} &= \varepsilon T_t^{\text{in}} + (1-\varepsilon)\left(T_t^{\text{out}} + \eta^{\text{AC}} \cdot \lambda \cdot P_t^{\text{AC}} \cdot \Delta T\right) \\ T_{\text{min}}^{\text{in}} &\leq T_t^{\text{in}} \leq T_{\text{max}}^{\text{in}} \end{aligned} \tag{3}$$

where T_t^{in} denotes the indoor temperature at time t; ε denotes the inertia factor; η^{AC} the thermal conversion efficiency; λ denotes the reciprocal of the thermal conductivity; ΔT denotes the scheduling resolution; $T_{\text{min}}^{\text{in}}$ and $T_{\text{max}}^{\text{in}}$ denote the minimum and maximum values of household preferred indoor temperatures, respectively.

The battery constraints restrict the temporal coupling of EV and ESS, which can be expressed as:

$$\begin{aligned} \text{SoC}_{t+1}^{\text{EV}} &= \text{SoC}_t^{\text{EV}} + \eta_{i,\text{cha}}^{\text{EV}} \cdot P_t^{\text{EV}} \cdot \Delta T / C_{\text{max}}^{\text{EV}} \\ \text{SoC}_{t+1}^{\text{ESS}} &= \begin{cases} \text{SoC}_t^{\text{ESS}} + \eta_{i,\text{cha}}^{\text{ESS}} \cdot P_t^{\text{ESS}} \cdot \Delta T / C_{\text{max}}^{\text{ESS}}, & P_t^{\text{ESS}} \geq 0 \\ \text{SoC}_{i,t}^{\text{ESS}} + \eta_{i,\text{dis}}^{\text{ESS}} \cdot P_t^{\text{ESS}} \cdot \Delta T / C_{\text{max}}^{\text{ESS}}, & P_t^{\text{ESS}} < 0 \end{cases} \\ \text{SoC}_{\text{min}}^{\text{EV}} &\leq \text{SoC}_t^{\text{EV}} \leq \text{SoC}_{\text{max}}^{\text{EV}} \\ \text{SoC}_{\text{min}}^{\text{ESS}} &\leq \text{SoC}_t^{\text{ESS}} \leq \text{SoC}_{\text{max}}^{\text{ESS}} \end{aligned} \tag{4}$$

where $\text{SoC}_{t+1}^{\text{EV}}$ and $\text{SoC}_{t+1}^{\text{EV}}$ denote the state of charge (SoC) of the EV and ESS, respectively; $\eta_{i,\text{cha}}$ denotes the charging efficiencies of the EV; $\eta_{i,\text{cha}}$ and $\eta_{i,\text{dis}}$ denote the charging and discharging efficiencies of the ESS, respectively; $C_{\text{max}}^{\text{EV}}$ and $C_{\text{max}}^{\text{ESS}}$ denotes the battery capacity of the EV and ESS, respectively; $\text{SoC}_{\text{min}}^{\text{EV}}$ and $\text{SoC}_{\text{max}}^{\text{EV}}$ denote the minimum and maximum SOC values of the EV, respectively; $\text{SoC}_{\text{min}}^{\text{ESS}}$ and $\text{SoC}_{\text{max}}^{\text{ESS}}$ denote the minimum and maximum SOC values of the ESS, respectively.

To save electricity costs C, storage systems and electric vehicles can be charged during off-peak tariff periods, while grid-connected electricity sales can be conducted during peak solar generation periods. However, due to prediction errors of P_t^{fcst}, smart meters may require further

adjustments to achieve the real-time supply and demand balance. In this case, a higher predicted load implies that consumers discard unused generated renewable energy, while a lower predicted load means that consumers have to temporarily purchase electricity in the energy market. Both situations are unfavorable for efficient energy management. In short, accurate forecasting results contribute to low additional grid electricity purchases and a high accommodation ratio of solar generation, thus reducing total carbon emissions. **The specific calculations for performance improvements are given as follows.**

First, we can obtain the day-ahead load forecasting results using the model trained by different edge intelligence methods. Thereafter, we can calculate the household electricity cost by solving the above forecasting-based optimization problem. The renewable energy accommodation ratio is defined as the ratio of the actual solar generation used for electricity supply to the total solar generation available. Furthermore, we can calculate the carbon emissions by multiplying the carbon emission intensity (CEI) by the electricity consumption. Note that the CEI for renewable energy generation is 0, while the CEI for grid-purchased power is 0.582 KgCO2/kWh in China [1]. Finally, we can calculate the reduced electricity cost, increased renewable energy accommodation, and reduced carbon emission by comparing the performance of the different methods.

We have added the following formulation of the home energy management problem in the revised supplementary information:

The objective of the short-term scheduling is to schedule the energy consumption of home appliances to help consumers reduce electricity cost based on forecasted load, which can be expressed as:

$$\min C = \sum_{t=1}^{\mathrm{T}} \lambda_{\mathrm{t}} \left(P_{\mathrm{t}}^{\mathrm{fcst}} + P_{\mathrm{t}}^{\mathrm{AC}} + P_{\mathrm{t}}^{\mathrm{EV}} + P_{\mathrm{t}}^{\mathrm{ESS}} - P_{\mathrm{t}}^{\mathrm{solar}} \right) \tag{5}$$

where T denotes the scheduling time scale; λ_t denotes the time-of-use electricity price; P_t^{fcst} denotes the forecasted load consumption; $P_{\mathrm{t}}^{\mathrm{AC}}$, $P_{\mathrm{t}}^{\mathrm{EV}}$, $P_{\mathrm{t}}^{\mathrm{ESS}}$, and $P_{\mathrm{t}}^{\mathrm{solar}}$ denote the power of air conditioner (AC), electric vehicle (EV), energy storage system (ESS), and solar panel, respectively. The positive and negative signs of $P_{\mathrm{t}}^{\mathrm{ESS}}$ correspond to the discharge and charging states of the ESS.

The feasibility constraints limit the operating power of appliances within a feasible range, which can be formulated as follows:

$$\begin{aligned} P_{\mathrm{min}}^{\mathrm{AC}} \leq P_{\mathrm{t}}^{\mathrm{AC}} \leq P_{\mathrm{max}}^{\mathrm{AC}} \\ P_{\mathrm{min}}^{\mathrm{EV}} \leq P_{\mathrm{t}}^{\mathrm{EV}} \leq P_{\mathrm{max}}^{\mathrm{EV}} \\ -P_{\mathrm{max}}^{\mathrm{ESS}} \leq P_{\mathrm{t}}^{\mathrm{ESS}} \leq P_{\mathrm{max}}^{\mathrm{ESS}} \end{aligned} \tag{6}$$

where $P_{\mathrm{min}}^{\mathrm{AC}}$ and $P_{\mathrm{max}}^{\mathrm{AC}}$ denote the minimum and maximum operating power of the AC, respectively; $P_{\mathrm{min}}^{\mathrm{EV}}$ and $P_{\mathrm{max}}^{\mathrm{EV}}$ denote the minimum and maximum charging power of the EV, respec-

tively; $P_{\text{max}}^{\text{ESS}}$ denote the maximum charging and discharging power of the ESS.

The thermal dynamics constraints restrict the indoor temperature within a comfortable range, which can be formulated as follows:

$$\begin{aligned} T_{t+1}^{\text{in}} &= \varepsilon T_t^{\text{in}} + (1-\varepsilon)\left(T_t^{\text{out}} + \eta^{\text{AC}} \cdot \lambda \cdot P_t^{\text{AC}} \cdot \Delta T\right) \\ T_{\text{min}}^{\text{in}} &\leq T_t^{\text{in}} \leq T_{\text{max}}^{\text{in}} \end{aligned} \tag{7}$$

where T_t^{in} denotes the indoor temperature at time t; ε denotes the inertia factor; η^{AC} the thermal conversion efficiency; λ denotes the reciprocal of the thermal conductivity; ΔT denotes the scheduling resolution; $T_{\text{min}}^{\text{in}}$ and $T_{\text{max}}^{\text{in}}$ denote the minimum and maximum values of household preferred indoor temperatures, respectively.

The battery constraint restricts the temporal coupling of EV and ESS, which can be expressed as:

$$\begin{aligned} \text{SoC}_{t+1}^{\text{EV}} &= \text{SoC}_t^{\text{EV}} + \eta_{i,\text{cha}}^{\text{EV}} \cdot P_t^{\text{EV}} \cdot \Delta T / C_{\text{max}}^{\text{EV}} \\ \text{SoC}_{t+1}^{\text{ESS}} &= \begin{cases} \text{SoC}_t^{\text{ESS}} + \eta_{i,\text{cha}}^{\text{ESS}} \cdot P_t^{\text{ESS}} \cdot \Delta T / C_{\text{max}}^{\text{ESS}}, & P_t^{\text{ESS}} \geq 0 \\ \text{SoC}_{i,t}^{\text{ESS}} + \eta_{i,\text{dis}}^{\text{ESS}} \cdot P_t^{\text{ESS}} \cdot \Delta T / C_{\text{max}}^{\text{ESS}}, & P_t^{\text{ESS}} < 0 \end{cases} \\ \text{SoC}_{\text{min}}^{\text{EV}} &\leq \text{SoC}_t^{\text{EV}} \leq \text{SoC}_{\text{max}}^{\text{EV}} \\ \text{SoC}_{\text{min}}^{\text{ESS}} &\leq \text{SoC}_t^{\text{ESS}} \leq \text{SoC}_{\text{max}}^{\text{ESS}} \end{aligned} \tag{8}$$

where $\text{SoC}_{t+1}^{\text{EV}}$ and $\text{SoC}_{t+1}^{\text{EV}}$ denote the state of charge (SoC) of the EV and ESS, respectively; $\eta_{i,\text{cha}}$ denotes the charging efficiencies of the EV; $\eta_{i,\text{cha}}$ and $\eta_{i,\text{dis}}$ denote the charging and discharging efficiencies of the ESS, respectively; $C_{\text{max}}^{\text{EV}}$ and $C_{\text{max}}^{\text{ESS}}$ denotes the battery capacity of the EV and ESS, respectively; $\text{SoC}_{\text{min}}^{\text{EV}}$ and $\text{SoC}_{\text{max}}^{\text{EV}}$ denote the minimum and maximum SOC values of the EV, respectively; $\text{SoC}_{\text{min}}^{\text{ESS}}$ and $\text{SoC}_{\text{max}}^{\text{ESS}}$ denote the minimum and maximum SOC values of the ESS, respectively.

[1] “Data Page: Carbon intensity of electricity generation”, part of the following publication: Hannah Ritchie, Pablo Rosado and Max Roser (2023) - “Energy”. Data adapted from Ember, Energy Institute. Retrieved from https://ourworldindata.org/grapher/carbon-intensity-electricity [online resource]

---

4. ‘To quantify the impact of edge intelligence on downstream ...’ I got confused here. From my understanding, successful energy management is based on the data/energy profile collected by the smart meters. Edge intelligence could be a way to realize energy management but not the only way. What is the role of the federated splitting learning-based edge intelligence here?

Reply:

Thanks for your insightful comment on the relationship between edge intelligence and energy management. We would like to provide further clarification on the role of federated split-

ting learning-based edge intelligence in energy management. Home energy management is an emerging technology that monitors, analyzes, and optimizes household energy consumption to reduce electricity costs [1]. This involves predicting future energy usage patterns from the data collected by smart meters to understand future consumption patterns and adjust behaviors accordingly. Traditional centralized analysis methods require the collection of fine-grained energy usage data from users, which may raise their privacy concerns. With strict privacy requirements, sensitive user data cannot be effectively utilized for any intelligent analysis.

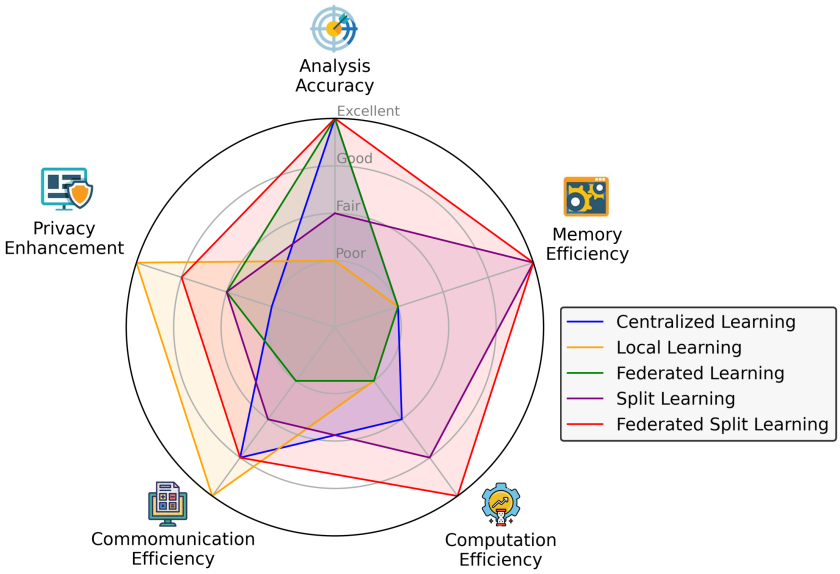


Fig. 1: Comparison among mainstream energy data analytics methods. The excellent, good, fair, and poor coordinate points represent the general performance of different methods in various dimensions. Our proposed federated split learning integrates the advantages of federated learning and split learning methods, achieving exceptional performance across all dimensions.

Edge intelligence is a promising solution to address these privacy concerns, as it allows smart meters to act as local agents for autonomous energy management by the users. However, conventional edge intelligence methods face challenges such as limited data and hardware resources. The federated split learning method combines the advantages of both split learning and federated learning to overcome these challenges. As shown in Fig. 1, our proposed approach can efficiently analyze energy data with higher accuracy while preserving user privacy, ultimately providing valuable insights for energy management.

In summary, the federated splitting learning-based edge intelligence plays a crucial role in offering an alternative solution for energy management that not only respects user privacy but also overcomes the limitations faced by traditional edge intelligence methods.

[1] Zhou B, Li W, Chan K W, et al. Smart home energy management systems: Concept, configurations, and scheduling strategies[J]. Renewable and Sustainable Energy Reviews, 2016, 61:

30-40.

Thanks again to all the reviewers for the valuable suggestions and comments. They helped us to improve the quality of the paper.