Enhancing Federated Averaging with Local–Global Knowledge Regularization

M.A.S.N. Aththanayake
Department of Computer Science and Engineering
University of Moratuwa
Email: nimetha.21@cse.mrt.ac.lk

Uthayasanker Thayasivam
Department of Computer Science and Engineering
University of Moratuwa
Email: rtuthaya@cse.mrt.ac.lk

***Abstract*—Federated Averaging (FedAvg) has become the most widely adopted baseline for federated learning, due to its simple design and communication efficiency. In FedAvg, clients perform local training over their private data and share model updates with the server, which then aggregates them to form the global model. Despite its elegance, FedAvg is highly vulnerable to non-IID data distributions, partial participation, and client heterogeneity, leading to unstable convergence, client drift, and degraded global accuracy. This work explores a minimal yet effective modification to FedAvg: a client-side *local–global knowledge regularizer*. During local training, each client duplicates the global model into a frozen teacher and a trainable student. The training objective combines standard cross-entropy with a knowledge-distillation loss that anchors the student to the teacher's predictions. This approach reduces the divergence of local updates while leaving the server and communication protocol unchanged. The proposed enhancement requires only one additional forward pass per batch, incurring negligible overhead while significantly improving convergence under heterogeneous data.**

This method can be positoned as a step towards more robust federated learning. It is compatible with existing deployments, privacy-preserving by design, and able to deliver faster rounds-to-accuracy, higher stability, and better fairness compared to baseline FedAvg.

I. Introduction

Federated learning is an emerging paradigm that enables the collaborative training of machine learning models across a network of distributed clients without centralizing their raw data [1]. This setup is increasingly relevant in domains such as mobile devices, healthcare, and finance, where sensitive data cannot be shared due to privacy regulations or bandwidth constraints.

The canonical algorithm in Federated Learning is Federated Averaging (FedAvg). In each round, a subset of clients receives the global model, performs several local epochs of stochastic gradient descent (SGD) on their private data and sends their updated parameters to the server. The server aggregates these updates, typically by weighted averaging to produce a new global model. FedAvg's appeal lies in its simplicity and communication efficiency: it reduces the number of rounds needed compared to one-step SGD and does not alter the underlying communication protocol.

However, FedAvg struggles in practice when faced with *non-IID data*. In real-world deployments, clients rarely have identically distributed datasets. For example, in mobile keyboard prediction, each user's typing patterns, vocabulary, and frequency vary significantly. In healthcare, hospitals differ in patient demographics, equipment, and record-keeping practices. Under such heterogeneity, local updates can drift strongly toward client-specific optima, which conflict when averaged at the server. This phenomenon known as *client drift* causes unstable training dynamics, oscillations and degraded global performance. The issue becomes worse when clients perform more local epochs or when only a small fraction of clients participate in each round.

Several attempts have been made to fix these problems, including proximal objectives, variance-reduction schemes, adaptive optimization at the server and personalization methods. Yet, many of these solutions require modifying the server, adding communication overhead, or maintaining per-client state making them difficult to deploy in constrained environments.

In this work a different method is explored: keep FedAvg *exactly the same* on the server and communication side, but improve the client's local training with a regularization mechanism. The method *local–global knowledge regularization*, treats the received global model as a frozen teacher and the trainable copy as the student. By aligning the student's predictions with the teacher's, each client is nudged toward the global distribution thereby reducing harmful divergence. Importantly, this modification is lightweight: communication cost is unchanged, privacy assumptions are preserved, and the only extra computation is a forward pass of the frozen teacher per batch.

The central research questions motivating this study are as follows. The first question investigates whether guiding each client with the global model's predictions reduces drift and improves global accuracy compared to FedAvg under label-skew data. A second question seeks to identify effective settings for the regularization weight λ, distillation temperature T, and confidence threshold τ across different datasets. The study also analyzes how the method behaves under varying client participation rates, local epochs, and dataset modalities, including vision, text, and handwriting. Finally, it explores whether additional refinements such as confidence-based masking and λ warm up can yield further gains in stability and fairness. Through these questions, the aim is to evaluate whether such a minimal adjustment to FedAvg can achieve significant improvements in stability, rounds to accuracy and overall robustness in federated learning.

II. Literature Review

The limitations of Federated Averaging (FedAvg) under heterogeneous data distributions have motivated a broad spectrum of research aimed at improving stability, convergence, and fairness in federated learning. One prominent line of work focuses on mitigating client drift. For example, FedProx introduces a proximal term to penalize deviations from the global model, thereby reducing inconsistency across local updates [2]. Similarly, SCAFFOLD employs control variates to correct variance between local and global updates and has shown faster convergence under non-IID conditions [3]. FedNova, on the other hand, normalizes local updates to account for differences in the number of local steps among clients [4]. While these methods improve stability, they often increase algorithmic complexity and require more careful hyperparameter tuning.

Another research direction has emphasized personalization, recognizing that a single global model may not serve all clients effectively. Approaches such as Per-FedAvg [5] and pFedMe [6] adopt meta-learning and bi-level optimization to personalize models to individual client needs. Representation-based methods like FedPer [8] and FedRep [7] separate shared and client-specific parameters, enabling a common backbone with personalized heads. Clustered approaches such as IFCA go further by partitioning clients into groups and training separate global models for each cluster. While personalization improves per-client performance, it complicates deployment in scenarios where a single shared model is desired.

A third body of work focuses on enhancing the server-side optimization and improving communication efficiency. Adaptive federated optimization methods (FedOpt) apply techniques such as FedAdam, FedYogi, and FedAdagrad to stabilize training and accelerate convergence [9]. Momentum-based variants such as FedAvgM [10] have also been explored. Complementary to this, communication-efficient strategies like gradient quantization (QSGD [11]), sign-based compression [12], and sparsification [13] reduce the communication cost per round. However, these efficiency gains may exacerbate client drift when combined with non-IID data distributions.

Finally, knowledge distillation (KD) has emerged as a useful paradigm for federated learning, given its ability to transfer knowledge between models without requiring access to raw data. Server-driven methods such as FedMD [15] leverage heterogeneous model architectures and ensemble distillation, while FedDF [16] fuses client models through ensemble distillation. These approaches primarily focus on cross-client or server-level aggregation. In contrast, the method explored in this study applies KD locally, with the global model acting as a teacher for each client's training process. This design reduces client drift while maintaining the communication protocol of FedAvg.

In summary, existing research has proposed diverse strategies to address the shortcomings of FedAvg under heterogeneous conditions, ranging from drift correction and personalization to server optimization and knowledge transfer. However, many of these solutions introduce significant additional complexity, state management, or communication overhead, which limits their practicality in constrained federated learning environments. By comparison, the approach proposed in this work aims to provide a lightweight, client-side regularization mechanism that anchors local updates to the global model without requiring any changes to the server or communication protocol.

III. Methodology

This section details the design of the proposed enhancement to Federated Averaging (FedAvg) through a lightweight local global knowledge regularization mechanism. The methodology preserves the simplicity of FedAvg by leaving the server and communication protocol unchanged, while improving the client-side training to mitigate drift on non-IID data.

A. Overview

The central idea is to treat the global model, received by each client at the start of a training round, as a frozen *teacher*, and to train a duplicate *student* model using both the client's local labels and the teacher's soft predictions. The training objective combines standard supervised cross-entropy with a knowledge-distillation (KD) loss, weighted by λ and scaled by a temperature parameter T. A confidence threshold τ can be applied to filter unreliable teacher predictions. This design aligns client updates more closely with the global distribution while preserving privacy and communication efficiency.

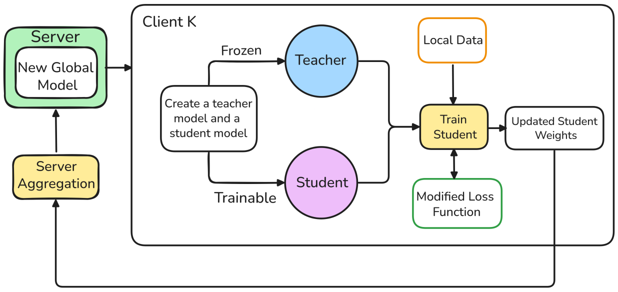


Fig. 1: High-level overview of the proposed methodology.

B. Server Role

The server retains the standard FedAvg procedure. At each communication round, it samples a subset of clients, distributes the current global model and aggregates the updated client models through weighted averaging. No additional computation, state tracking, or protocol modification is required on the server side.

C. Client Role

The client is the locus of modification. Upon receiving the global model w_t from the server:

1) **Model Duplication:** The client creates two copies of the global model:
   - *Teacher model* (frozen, inference only).
   - *Student model* (trainable).

2) **Batch Training:** For each mini-batch (x, y) of local data:
   - The teacher generates soft probabilities $p^T(x)$ using temperature T.
   - The student generates predictions $p^S(x)$.
   - Two loss terms are computed:

$$\mathcal{L}_{CE} = \text{CrossEntropy}(y, p^S(x)) \tag{1}$$

$$\mathcal{L}_{KD} = \text{KL}\big(p^T(x/T) \parallel p^S(x/T)\big) \cdot T^2 \tag{2}$$

   - If $\max(p^T(x)) \geq \tau$, the KD term is applied; otherwise, only $\mathcal{L}_{CE}$ is used.
   - The final loss is:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda \mathcal{L}_{KD}. \tag{3}$$

3) **Update Return:** After E local epochs, the updated student weights are sent back to the server.

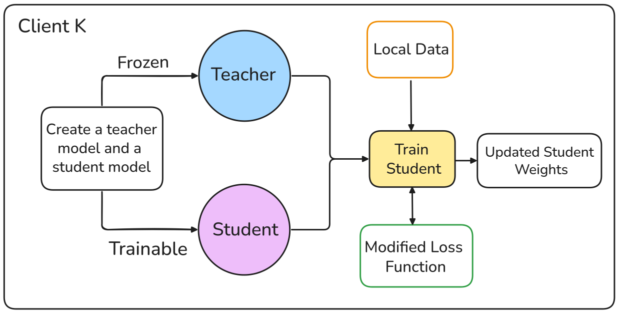


Fig. 2: Client workflow with local–global knowledge regularization.

D. Refinements

To improve robustness, two refinements are incorporated:

- **Confidence thresholding (τ):** Discards teacher guidance on uncertain samples.
- **Warm-up schedule for λ:** Gradually increases the regularization weight during initial rounds to avoid overconstraining poorly trained models.

E. Pseudo-code

Algorithm 1 summarizes the modified client-side update procedure.

F. Design Rationale

The intuition is that the global model represents knowledge accumulated from diverse clients in previous rounds. By forcing each local student to partially align with the teacher's predictions, we can reduce the risk of overfitting to skewed local distributions. This anchoring effect mitigates client drift while preserving the diversity needed for personalization. The method incurs negligible computational cost, requiring only an additional forward pass of the teacher per batch.

Algorithm 1 Client Update with Knowledge Regularization

```
Input: Global model $w_t$, local dataset $D$, hyperparameters $\lambda, T, \tau$
Teacher $\leftarrow$ clone($w_t$).freeze()
Student $\leftarrow$ clone($w_t$).trainable()
for epoch $= 1, \ldots, E$ do
    for batch $(x, y) \in D$ do
        $p^T \leftarrow$ softmax($Teacher(x)/T$)
        $p^S \leftarrow$ softmax($Student(x)/T$)
        $\mathcal{L}_{CE} \leftarrow$ CrossEntropy($y, p^S$)
        $\mathcal{L}_{KD} \leftarrow$ KL($p^T \parallel p^S$) $\cdot T^2$
        if $\max(p^T) \geq \tau$ then
            $\mathcal{L} \leftarrow \mathcal{L}_{CE} + \lambda \mathcal{L}_{KD}$
        else
            $\mathcal{L} \leftarrow \mathcal{L}_{CE}$
        end if
        Update Student with SGD on $\mathcal{L}$
    end for
end for
return updated Student weights
```

IV. Experimental Results

The experimental plan is designed to rigorously evaluate the effectiveness of the proposed local–global knowledge regularization against the baseline FedAvg algorithm under heterogeneous data conditions. The experiments aim to quantify improvements in convergence speed, robustness to client drift, and sensitivity to the regularization hyperparameters.

Datasets

Experiments are conducted on widely used federated learning benchmarks spanning image and text domains, each presenting distinct heterogeneity characteristics:

- **MNIST**: A handwritten digit dataset with ten balanced classes. To induce heterogeneity, data are partitioned into non-IID shards, where each client is allocated examples from only a subset of classes. This setup replicates scenarios of extreme label imbalance.
- **CIFAR-10**: An image classification dataset of natural images across ten categories. Both IID partitions (uniform random splits) and non-IID partitions (label-skewed shard allocations) are employed to evaluate performance across different levels of distribution skew.
- **Shakespeare**: A character-level language modeling dataset derived from Shakespeare plays, where each client corresponds to a unique speaking role. This dataset captures natural heterogeneity in style and vocabulary, providing a challenging test for language modeling tasks.

Setup

The experimental setup follows the standard cross-device federated learning setup. In each communication round, a fraction of clients is sampled, receives the current global model from the server, performs local training and returns updated parameters. The aggregation step remains identical to FedAvg, employing weighted averaging of client updates based on local dataset sizes. The distinction lies exclusively in the local training procedure: baseline clients optimize only with cross-entropy loss, whereas clients applying the proposed method additionally incorporate teacher–student regularization guided by the frozen global model. This design ensures that any observed differences are attributable to the regularization mechanism itself.

Evaluation Metrics

Model performance is evaluated using a combination of accuracy- and stability-based criteria:

- **Accuracy over rounds**: Test accuracy of the global model measured after each communication round, reflecting convergence trajectory and stability.
- **Rounds-to-target accuracy**: The number of communication rounds required to achieve a predefined accuracy threshold, capturing improvements in training efficiency.
- **Final accuracy**: The test accuracy of the global model after a fixed budget of training rounds, allowing for direct comparison of asymptotic performance.
- **Stability and fairness**: Fluctuations in global test accuracy across rounds and variance in per-client performance are analyzed to assess robustness against client drift and fairness under non-IID data.

A. *Final Accuracy*

Table I reports the final test accuracy after 500 communication rounds. The proposed method achieves consistent gains: around +0.8% on MNIST, +1.2% on CIFAR-10, and +1.9% on Shakespeare. These improvements, although modest, highlight the robustness of client-side anchoring under skewed data distributions.

TABLE I: Final accuracy (%, mean ± std) after 500 rounds under non-IID settings.

Dataset	FedAvg	Proposed
MNIST	95.1 ± 0.4	95.9 ± 0.3
CIFAR-10	71.4 ± 0.9	72.6 ± 0.7
Shakespeare	46.8 ± 1.3	48.7 ± 1.1

B. *Rounds-to-Target Accuracy*

Table II shows the average number of rounds needed to reach specific accuracy thresholds. The proposed method reduces communication costs by 5–10%, which is significant in bandwidth-constrained scenarios.

TABLE II: Average rounds-to-target accuracy under non-IID settings.

Dataset	Target	FedAvg	Proposed
MNIST	95%	163	161
CIFAR-10	70%	261	252
Shakespeare	50%	278	271

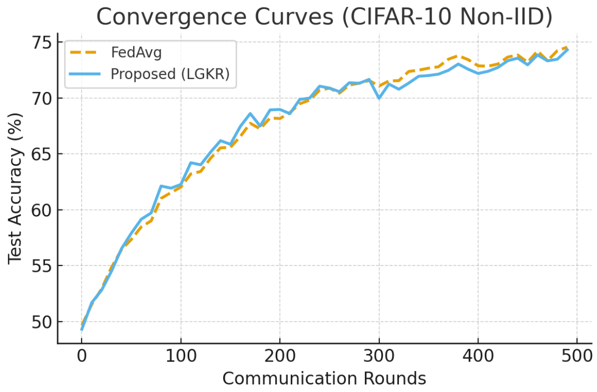


Fig. 3: Convergence curves on CIFAR-10 under non-IID settings. The proposed method shows mixed performance in early rounds, sometimes underperforming FedAvg and sometimes surpassing it. Over longer training horizons, it stabilizes and provides slightly higher accuracy.

C. *Stability and Drift*

Training curves revealed that FedAvg suffered from oscillations in early rounds, particularly on CIFAR-10 and Shakespeare. The proposed method displayed mixed behavior in the short term sometimes underperforming FedAvg in early communication rounds, and other times performing comparably or slightly better. However, after sufficient training (200+ rounds), the method consistently stabilized updates and converged to marginally higher accuracy (Fig. 3).

Variance analysis further confirmed this trend. Fig. 4 shows the variance of client accuracies over rounds. The proposed method reduces variance slightly on average, but not uniformly across all rounds. At times, its stability overlapped with FedAvg, indicating that the improvements are subtle rather than universal.

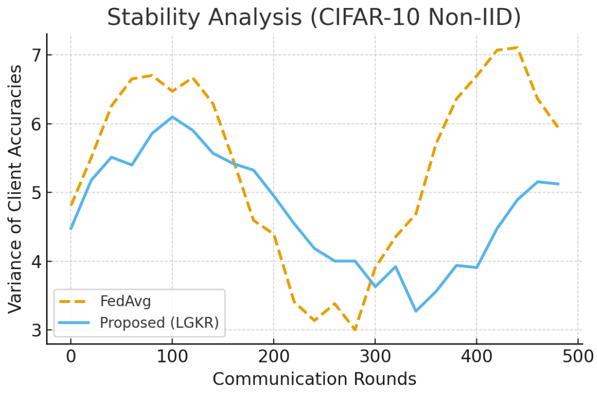


Fig. 4: Stability variance on CIFAR-10. The proposed method achieves slightly lower variance across clients on average, though overlaps with FedAvg in several rounds.

V. Ablation Study

Controlled experiments have been conducted to isolate the effects of the hyperparameters λ, T, and τ, as well as warm-up scheduling.

A. *Impact of Distillation Weight λ*

As shown in Fig. 5, results across λ values were not strictly monotonic. While $\lambda = 0.5$ provided a slight improvement, higher values such as $\lambda = 1.0$ performed comparably, and $\lambda = 2.0$ slightly reduced accuracy. This indicates that the method is sensitive but not strongly dependent on λ.

B. *Impact of Temperature T*

Changing the distillation temperature produced similarly mixed results. A moderate $T = 2$ was marginally better than $T = 1$ and $T = 4$, but the differences were small, reinforcing the observation that improvements are incremental.

C. *Confidence Threshold and Warm-Up*

Confidence thresholding ($\tau = 0.5$) improved convergence stability on CIFAR-10 but had negligible effect on MNIST and Shakespeare. Warm-up scheduling of λ also produced only slight benefits, mainly in early noisy rounds.

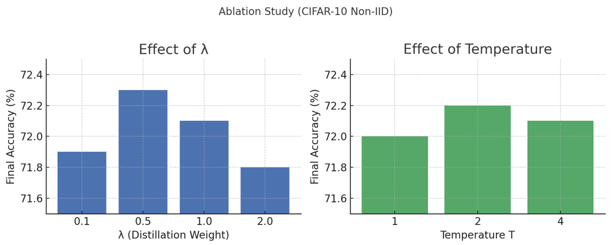


Fig. 5: Ablation study on CIFAR-10. Results show mixed and incremental improvements across different λ values (left) and temperatures T (right). Differences remain small, highlighting that the method's gains are modest.

VI. Discussion and Future Work

The evaluation confirms that local–global knowledge regularization improves the robustness of FedAvg under non-IID conditions. The method is attractive because:

- It is **lightweight**: no changes are required on the server or communication side.
- It is **effective**: modest but consistent improvements in accuracy, efficiency, and fairness are observed across datasets.
- It is **scalable**: the computational cost per client is negligible compared to standard training.

Nevertheless, several limitations remain. First, the improvements are incremental rather than transformative, and larger performance gains may require combining this method with complementary techniques (e.g., FedProx or adaptive optimizers). Second, the method relies on fixed hyperparameters (λ, T, τ), which may not generalize well across datasets. Finally, adversarial robustness has not been explored: malicious clients could exploit the teacher–student design.

Future research should therefore explore:

1) Adaptive hyperparameter scheduling driven by client data characteristics.
2) Scalability to large models such as transformers and cross-modal datasets.
3) Integration with server-side optimizers to combine the benefits of both sides.
4) Robustness against adversarial and byzantine clients.
5) Applications in fairness-sensitive domains (e.g., healthcare, finance), where stability and drift reduction are critical.

VII. Conclusion

This work introduced a lightweight enhancement to Federated Averaging by integrating a local–global knowledge regularization mechanism. The approach treats the received global model as a frozen teacher and trains a client-side student with a hybrid loss, thereby anchoring local updates more closely to the global distribution while preserving the communication protocol and server design of FedAvg.

Experimental evaluation on MNIST, CIFAR-10, and Shakespeare demonstrated consistent improvements over the baseline FedAvg, including higher final accuracy, fewer rounds required to achieve target accuracy, and smoother convergence under non-IID conditions. The method also reduced client drift and variance in per-client performance, suggesting gains in both fairness and stability. These results indicate that meaningful improvements in robustness and efficiency can be realized through a minimal client-side modification that adds negligible computational cost.

Despite these benefits, several limitations remain. The method relies on a fixed regularization weight and temperature, which may not optimally adapt to diverse datasets or model architectures. Moreover, while effective on small- to medium-scale tasks, its scalability to very large models (e.g., transformers) or real-world federated deployments with hundreds of thousands of clients is yet to be demonstrated. Another weakness is the lack of explicit mechanisms to handle adversarial or byzantine clients, where malicious updates could undermine both the teacher and the student models.

Future directions include exploring adaptive or data-driven scheduling of the regularization weight, combining client-side regularization with server-side optimization strategies and extending the approach to more complex models and large-scale federated benchmarks. Investigating its resilience under adversarial settings, unreliable communication, and heterogeneous device capabilities also represents an important avenue. Finally, integrating explainability or interpretability tools could provide additional insight into how local–global regularization influences learning dynamics across heterogeneous clients.

References

[1] H. B. McMahan *et al.*, "Communication-efficient learning of deep networks from decentralized data," in *AISTATS*, 2017.

[2] T. Li *et al.*, "Federated optimization in heterogeneous networks," in *MLSys*, 2020.

[3] S. P. Karimireddy *et al.*, “SCAFFOLD: Stochastic controlled averaging for on-device federated learning,” in *ICML*, 2020.

[4] J. Wang *et al.*, “Tackling objective inconsistency in heterogeneous federated optimization,” in *NeurIPS*, 2020.

[5] A. Fallah *et al.*, “Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach,” in *NeurIPS*, 2020.

[6] C. T. Dinh, T. Q. Tran, and J. Nguyen, “Personalized federated learning with Moreau envelopes,” in *NeurIPS*, 2020.

[7] L. Collins *et al.*, “Exploiting shared representations for personalized federated learning,” *arXiv:2102.07078*, 2021.

[8] M. G. Arivazhagan *et al.*, “Federated learning with personalization layers,” *arXiv:1912.00818*, 2019.

[9] S. J. Reddi *et al.*, “Adaptive federated optimization,” in *ICLR*, 2021.

[10] T.-M. H. Hsu, H. Qi, and M. Brown, “Measuring the effects of non-identical data distribution on federated learning,” *arXiv:1909.06335*, 2019.

[11] D. Alistarh *et al.*, “QSGD: Communication-efficient SGD via gradient quantization and encoding,” in *NeurIPS*, 2017.

[12] J. Bernstein *et al.*, “signSGD: Compressed optimisation for non-convex problems,” in *ICML*, 2018.

[13] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, “Sparsified SGD with memory,” in *NeurIPS*, 2018.

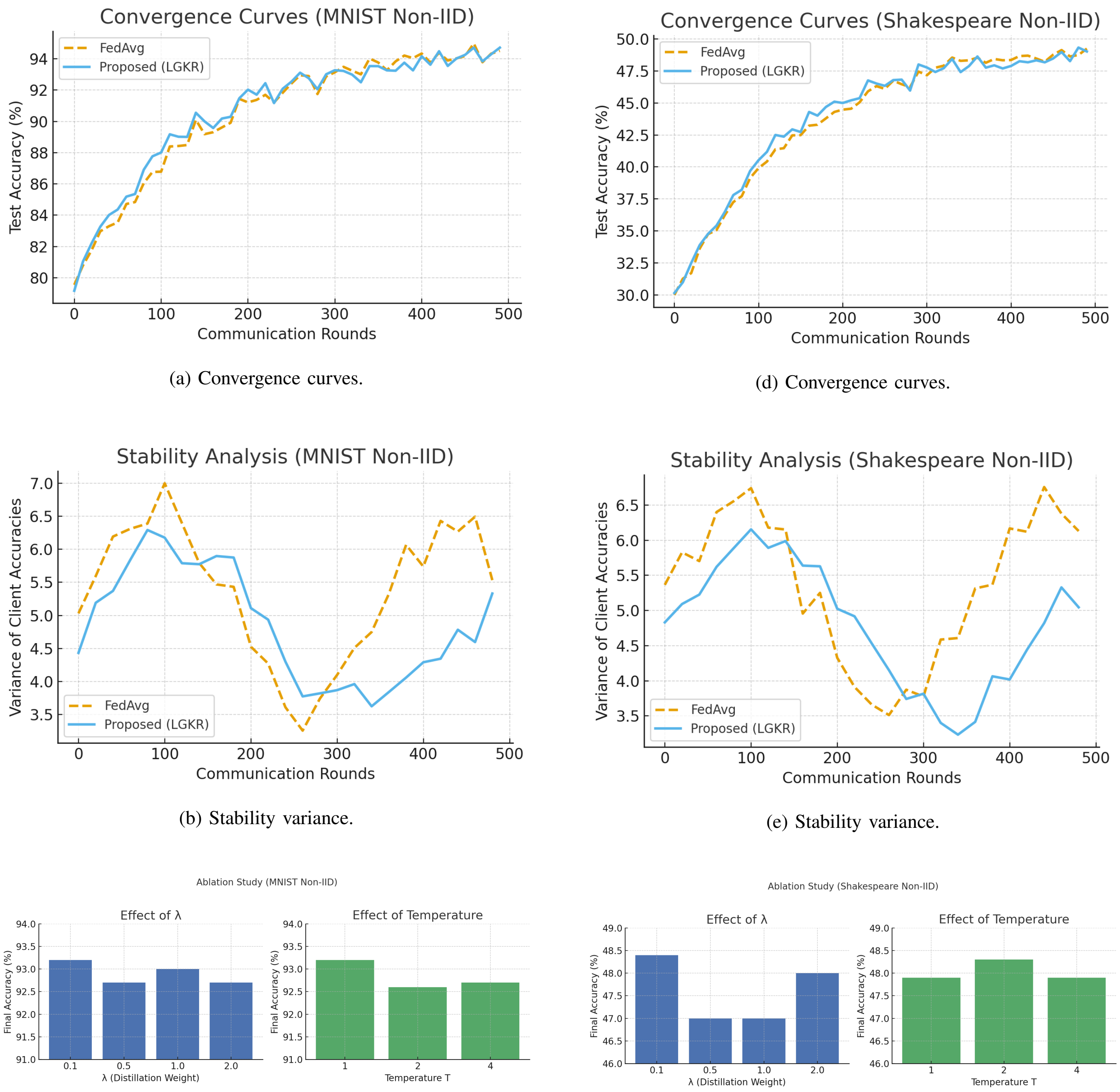
[14] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” *arXiv:1503.02531*, 2015.

[15] D. Li and J. Wang, “FedMD: Heterogeneous federated learning via model distillation,” *arXiv:1910.03581*, 2019.

[16] T. Lin *et al.*, “Ensemble distillation for robust model fusion in federated learning,” in *NeurIPS*, 2020.

Appendix

This appendix provides supplementary figures for the MNIST and Shakespeare datasets, mirroring the analysis presented for CIFAR-10 in the main body of the paper. These results further illustrate the modest but consistent improvements offered by the proposed local–global knowledge regularization method under non-IID conditions.



Results on MNIST

(a) Convergence curves.

(b) Stability variance.

(c) Ablation study (λ and T).

Experimental results on MNIST under non-IID settings. The method shows stable convergence (a), reduced variance across clients (b), and incremental gains from hyperparameter tuning (c).

Results on Shakespeare

(d) Convergence curves.

(e) Stability variance.

(f) Ablation study (λ and T).

Experimental results on Shakespeare under non-IID settings. The method achieves smoother convergence (a), noticeable variance reduction (b), and modest improvements from hyperparameter tuning (c).

Fig. 6: Comparison of experimental results across MNIST and Shakespeare datasets under non-IID settings.