Appendix

A Notations

All notations used in this paper are summarized in Table 4. We omit the superscript of symbols indicating temporal information, e.g., $\mathcal{J}^n$ refers to a set of participating clients in communication round n.

Table 4: Symbols and notations in the paper.

Symbol	Explanation
$\mathcal{J}$	Set of clients
$\mathcal{D}_j, \hat{\mathcal{D}}_j$	Set of local/representative samples of client j
$\tilde{\mathcal{D}}_j$	Set of generated samples for client j
$\tilde{\mathcal{T}}_j, \mathcal{T}_j$	Tokens sequences/text captions from client j
$\mathcal{G}_s$	Data pool maintained by server
$\mathcal{M}_i$	Set of text prompts for class i
$\mathcal{W}_j$	Weights of updated local model
$\bar{\mathcal{W}}, \mathcal{W}$	Weights of aggregated/fine-tuned global model
$\mathbf{D}$	Decision matrix for data generation
$\mathcal{I}$	Set of classes of generated samples
τ_j	Number of local SGD steps for client j
N	Number of communication rounds
C	Number of task-specific classes
G	Budget for the number of generated samples
T_v, T_s	Threshold for validation accuracy/text similarity

B Experimental Details

B.1 Visualization of the PACS and Office-Caltech dataset

We evaluate the performance of Flick and its counterparts on two benchmark image datasets: PACS [25] and Office-Caltech [44]. Each benchmark is a multi-class classification task, where each class includes samples from different domains. The PACS dataset comprises seven classes: *dog, elephant, giraffe, guitar, horse, house, and person*. The Office-Caltech dataset contains ten classes:

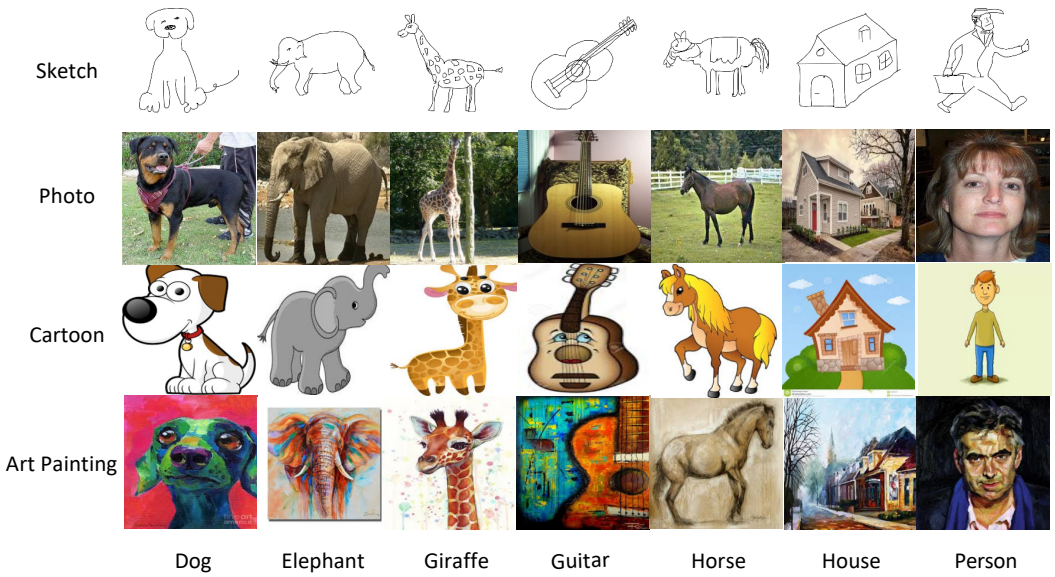


Figure 7: Example samples from different classes and domains in the PACS dataset [25].

Figure 8: Example samples from different classes and domains in the Office-Caltech dataset [44].

backpack, bike, calculator, headphones, keyboard, laptop, monitor, mouse, mug, and projector. Figures 7 and 8 illustrate the diverse domains within individual classes in the two datasets. In the PACS dataset, significant feature variations across domains within the same class are clearly observable, making domain differences explicit. In contrast, domain differences in Office-Caltech are more latent, for instance, variations with background, view, and clarity of the object, posing additional challenges in effectively addressing the domain shift problem within this benchmark. To simulate heterogeneous FL settings with domain shifts, we partition the data and assign data from one domain exclusively to each client.

B.2 Hyperparameter Settings

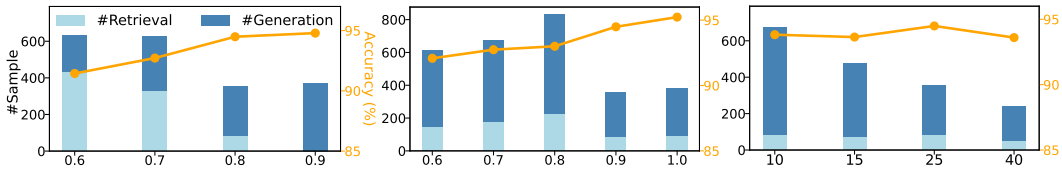
Here, we provide more details about the experimental training parameter settings. All methods are implemented in Python, with neural networks developed using PyTorch. In local training, local epochs are set to 5, and the input data size is fixed at 224×224. We set the server-held data pool size $|\mathcal{G}_s|$ to 25, and the data generation budget G to 5 for both datasets. We use "stable-diffusion-v1-5" for the image generator and "gpt-4o-mini" and out-of-the-box LLMs for generating text prompts by analyzing the uploaded client-specific knowledge combined with embedded commonsense knowledge. A comprehensive robustness evaluation of Flick under different image generators and out-of-the-box LLMs is provided in Appendix C.1. Additionally, for the PACS dataset, we set text similarity threshold $T_s = 0.8$ and validation threshold $T_v = 0.9$, which guide the construction of the decision matrix $\mathbf{D}$. For the Office-Caltch dataset, we adopt $T_s = 0.7$ and $T_v = 0.8$ as the default settings.

In baseline methods, FedProx uses a hyperparameter to control the proximal term's weight in the objective function. We tune this hyperparameter within the $[0.001, 0.01, 0.1, 1]$ following [9], and report the best result. Specifically, the value of 0.1 is adopted in both datasets. For FedDyn, we provide its best performance by fine-tuning the weight of penalized risk within $[0.001, 0.01, 0.1]$. We set this value to 0.1 for PACS and 0.01 for Office-Caltech. For FGL, we adopt the multi-round-syn variant, where the global model is initially trained on a synthetic dataset generated in the first round and subsequently fine-tuned using synthetic data throughout the following communication rounds. To ensure a fair comparison, we match the total amount of synthetic data to that produced by our method, Flick, under each experimental setting. For other counterparts, we keep the default settings they provided.

C More Results and Analysis

C.1 Sensitivity Analysis

In this section, we demonstrate that Flick consistently maintains superior performance across a range of hyperparameter settings and model choices, indicating the robust generalization of Flick without reliance on extensive tuning.



(a) Text similarity threshold T_s (b) Validation threshold T_v (c) Data pool size

Figure 9: The performance of Flick under varying hyperparameters on the PACS dataset.

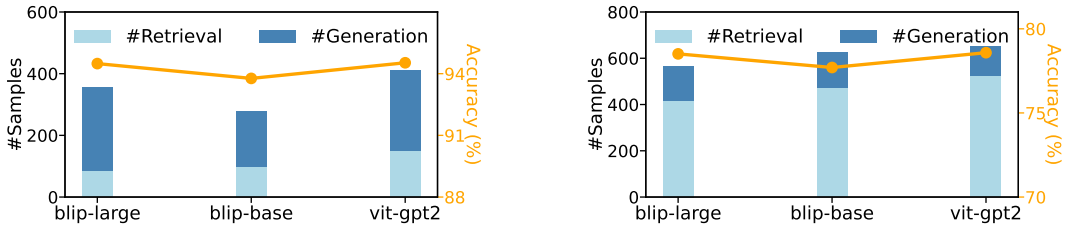
Impact of text similarity T_s, validation accuracy T_v, and data pool size. Figure 9(a) shows the model performance improvements with a larger text similarity threshold T_s. The number of generated samples required to reach the target accuracy (i.e., FedAvg's best accuracy) rises with a higher T_s, while #Sample decreases because fewer rounds are needed. When T_s reaches 0.9, no samples are retrieved. For validation accuracy threshold T_v, a lower value generates fewer samples per round but requires more rounds to reach the target accuracy. With a fixed number of rounds, the accuracy is lower at smaller values of T_v. As shown in Figure 9(b), $T_v = 0.9$ or 1.0 serves as an effective threshold, yielding better performance with fewer required samples. Figure 9(c) shows that the accuracy performance does not change much across different sizes of the server-held data pool. Here, x-axis denotes storage capacity per class in the data pool; we can observe that a larger pool size significantly benefits the global model's fine-tuning, greatly reducing #Sample. However, a larger data pool size requires more storage.

Impact of Data Generation Budget G. In our designed prompt illustrated in Figure 3, the server hosting the LLM can generate text prompts of varying sizes for class i under a specified hyperparameter G. This design flexibility in our Flick allows the server to adapt to the practical deployment with the given data generation budget. In the experiments, we evaluate the performance of Flick under different data generation budgets $G \in \{5, 10, 15\}$. We keep the experimental setup for the two benchmarks consistent with the experiments in Table 1, with FedAvg as the baseline method, except for varying G and the server-held data pool size fixed at $|\mathcal{G}_s| = 40$. The results are shown in Table 5. We can observe that a larger G indeed improves the global model accuracy in the long run by generating more samples enriched with commonsense knowledge. Additionally, $G = 10$ and $G = 15$ speed up the model convergence on the Office-Caltech dataset compared to $G = 5$, as evidenced by the enhanced round-to-accuracy performance (i.e., #Round). However, $G = 5$ requires the fewest generated samples (i.e., #Sample) to reach the target accuracy, revealing a trade-off between training performance (e.g., accuracy and latency) and data generation overheads. It underscores the importance of carefully selecting the G to balance data generation costs and performance improvements in practical applications. In the rest of the evaluations in this paper, we use $G = 5$ as the default setting.

Table 5: The performance of our proposed Flick under different data generation budget G on both datasets.

G	PACS						
	P	A	C	S	AVG ↑	#Round ↓	#Sample ↓
$G = 5$	91.99	91.51	97.62	93.27	93.60±0.17	7	188
$G = 10$	92.72	94.06	96.43	93.65	94.21±0.20	7	276
$G = 15$	93.93	93.84	97.17	93.53	94.62±0.14	7	420
G	**Office-Caltech**						
	A	C	D	W	AVG ↑	#Round ↓	#Sample ↓
$G = 5$	76.68	65.78	88.10	80.56	77.78±0.12	33	146
$G = 10$	77.98	63.56	94.64	79.17	78.84±0.22	24	253
$G = 15$	77.72	65.33	96.43	78.33	79.45±0.18	24	317

Impact of different image captioning methods. In Flick, clients perform image captioning on a small set of representative local samples during the summary phase. This summarization captures essential local insights that play a crucial role in guiding the server to extract task-relevant, high-quality commonsense knowledge from LLMs – an effect validated in our ablation study. To assess the sensitivity of Flick to different image captioning strategies, we evaluate its performance using

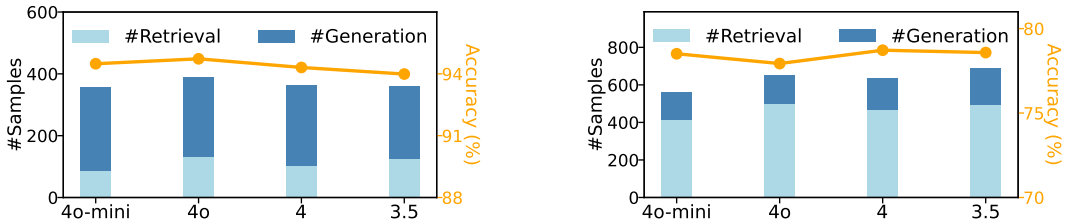


(a) PACS dataset (b) Office-Caltech dataset

Figure 10: The performance of Flick over different image captioning methods: blip-image-captioning-large, blip-image-captioning-base, and vit-gpt2-image-captioning.

three representative captioning models: “blip-image-captioning-large", “blip-image-captioning-base", and “vit-gpt2-image-captioning". As shown in Figure 10, the results on both datasets indicate that Flick is agnostic to the choice of captioning model and consistently maintains superior performance. This demonstrates the flexibility and generalization of our approach.

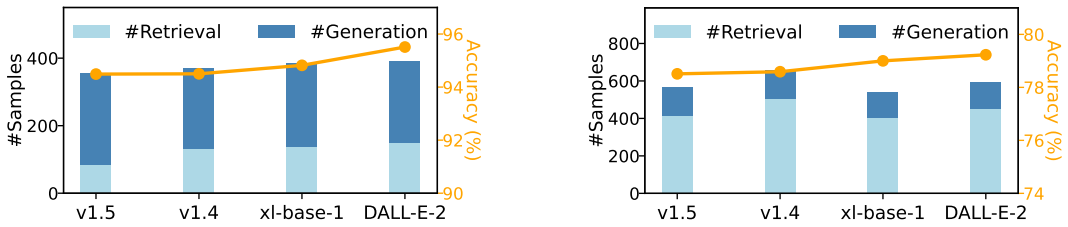
Impact of different out-of-the-box LLMs. The server in Flick employs LLM to analyze the collected local summary and generate text prompts for subsequent data generation. The designed prompt template instructs the LLM to extract class-specific information from collected local captions and fuse it with inherent commonsense knowledge, thereby enhancing the quality of the output text prompts. To demonstrate the compatibility of the Flick workflow with out-of-the-box LLMs, we use the same prompt template as illustrated in Figure 3 and evaluate the performance of Flick by configuring the LLM to “gpt-3.5-turbo-0125”, “gpt-4”, “gpt-4o”, and “gpt-4o-mini” (default LLM in this paper). The experimental results are given in Figure 11. We can see that the choice of LLM has a negligible impact on the text prompt generation, evidenced by the similar global model performance and data generation decisions. These valuation results validate the compatibility of our designed prompt template across a wide range of LLMs.



(a) PACS dataset (b) Office-Caltech dataset

Figure 11: The performance of Flick over different out-of-the-box LLMs: gpt-4o-mini, gpt-4o, gpt-4, and gpt-3.5-turbo-0125.

Impact of Image Generator $f^{\mathcal{W}_G}(\cdot)$. As a key component of the Flick workflow, a text-to-image model is employed to generate synthetic samples taking input text prompts. Since the data generation runs on the server side, the availability of rich computational resources allows for extending the choice within advanced generative image models, thereby enabling the generation of high-quality data points for further usage in Flick. To this end, Figure 12 reports the performance of Flick when using various generative models. We investigate the LDMs publicly available in HuggingFace, including “stable-diffusion-v1-5” (default model in this paper), “stable-diffusion-v1-4”, and “stable-diffusion-xl-base-1.0”. We also incorporate the “DALL-E-2” embedded in ChatGPT/OpenAI API for comparison. The results indicate that synthetic samples provided by all four image generators significantly enhance FL performance compared to the baseline FedAvg that achieves 83.06% on PACS and 70.84% on Office-Caltech. Note that the models with more advanced designs improve the global model’s performance better, at the cost of higher computational demands. E.g., “DALL-E-2” model leads to higher accuracy, potentially due to its ability to produce samples in diverse artistic styles [53].



(a) PACS dataset (b) Office-Caltech dataset

Figure 12: The performance of Flick over different image generators: stable-diffusion-v1-5, stable-diffusion-v1-4, stable-diffusion-xl-base-1.0, and DALL-E-2.

C.2 Extended Evaluation on Flick Variants

In Table 3, we conduct the ablation study on Flick by evaluating its performance without the *Local Summary (LS)* phase or the *Data Retrieval (DR)* during the data generation phase. To further explore the effectiveness of our LS and DR designs–specifically the loss-based LS and similarity-based DR strategies–we compare Flick with its variants: random local sample captioning (*Random Captioning*) and random historical sample retrieval (*Random Retrieval*). Table 6 summarizes the results under the setup of FedAvg as the baseline method for both benchmarks. For a fair comparison, the server in the *Random Retrieval* retrieves the same number of samples as Flick but randomly selects from the server-held data pool. This ensures that *Random Retrieval* maintains the same proportion of retrieved samples $\mathcal{D}\text{retr}$ and generated samples $\mathcal{D}\text{gen}$ as Flick during the entire data generation phase across all communication rounds. The experimental results clearly reveal the benefits of selective local sample summarization and guided historical sample retrieval in Flick over their random counterparts. Random local sample captioning results in inefficient extraction of client-specific knowledge, as it lacks the focus on captioning the representative local samples provided by the loss-based criteria. Furthermore, the negative impact of introducing randomness into data retrieval is even more severe in *Random Retrieval*, where the central server randomly sends diverse synthetic samples that fail to align with the specific requirements of local data compensation, thereby degrading the training efficiency.

Table 6: Extended ablation study on the components of client-side local summary and server-side data generation in Flick. We compare Flick with its variants: Flick with random local sample captioning (*Random Captioning*) and Flick with random historical sample retrieval (*Random Retrieval*).

Methods	**PACS**						
	P	A	C	S	AVG ↑	#Round ↓	#Sample ↓
Random Retrieval	89.81	91.40	96.28	93.21	92.67±0.07	19	423
Random Captioning	91.87	91.61	97.32	93.08	93.47±0.33	19	430
Flick	91.99	94.59	97.74	93.91	**94.49**±0.10	**11**	**270**
Methods	**Office-Caltech**						
	A	C	D	W	AVG ↑	#Round ↓	#Sample ↓
Random Retrieval	78.24	65.78	89.29	73.33	76.66±0.52	45	295
Random Captioning	75.13	61.33	92.86	79.17	77.12±0.20	43	235
Flick	75.82	64.00	96.43	77.78	**78.51**±0.56	**37**	**150**

C.3 Detailed Evaluation under Domain Shift

The data generation process in Flick considers the class-wise performance of local models (i.e., through the decision matrix $\mathbf{D}$) and the domain knowledge variations across clients (i.e., the collection of local summaries $\mathcal{T}$ used in LLM prompt). This highlights the capability of Flick to solve domain shift and label skew in heterogeneous FL jointly and therefore distinguishes Flick from other model-driven [43, 16, 9] or data-driven [17, 18, 21, 22] methods that solely address imbalanced local datasets across clients. To validate the performance gain by Flick attributed to mitigating domain shift, Table 7 reports experimental results where each client holds balanced local data in label

Table 7: Global model accuracy performance (%) under domain shift only. AVG denotes the average accuracy calculated across all domains, and Δ indicates the accuracy gain compared with vanilla methods. **Best** in bold. *Acronyms in the PACS dataset: Photo (P), Art Painting (A), Cartoon (C), and Sketch (S); In the Office-Caltech dataset: Caltech (C), Amazon (A), Webcam (W), and DSLR (D).*

Methods	PACS							Office-Caltech						
	P	A	C	S	AVG	$\Delta\uparrow$	#Round $\downarrow$	A	C	D	W	AVG	$\Delta\uparrow$	#Round $\downarrow$
FedAvg [5]	91.85	94.16	95.98	87.88	92.47$\pm$0.11	-	84	77.47	63.33	70.37	88.56	74.93$\pm$0.70	-	59
+FedBN [28]	89.53	93.35	95.63	90.91	92.36$\pm$0.1	-0.13	-	77.60	64.22	72.22	88.98	75.76$\pm$0.35	0.83	45
+FedHEAL [27]	91.00	94.69	95.09	91.37	93.04$\pm$0.22	0.57	79	76.43	65.11	75.93	91.10	77.14$\pm$1.18	2.21	38
+DynaFed [29]	90.27	92.71	95.93	89.68	92.15$\pm$0.45	-0.32	-	79.69	65.33	70.37	84.75	75.03$\pm$0.94	0.10	55
+FedFTG [19]	89.78	92.99	90.77	92.39	91.48$\pm$0.63	-0.99	-	82.29	65.00	67.96	78.03	73.32$\pm$0.90	-1.61	-
+FGL [20]	89.90	90.13	95.09	89.97	91.27$\pm$0.05	-1.20	-	61.04	62.22	85.19	81.19	72.41$\pm$0.40	-2.52	-
+Flick	91.48	94.48	96.88	92.20	**93.76**$\pm$0.07	**1.29**	**24**	81.60	65.48	87.65	91.53	**81.56**$\pm$0.54	**6.63**	**21**
FedProx [9]	88.56	93.63	94.05	91.50	91.94$\pm$0.25	-	84	78.52	62.56	75.00	90.68	76.69$\pm$0.34	-	59
+FedBN [28]	89.05	93.74	94.94	89.59	91.83$\pm$0.14	-0.11	-	74.35	64.11	78.70	88.56	76.43$\pm$1.65	-0.26	-
+FedHEAL [27]	90.88	95.12	95.39	90.99	93.09$\pm$0.11	1.15	41	79.95	67.33	74.07	91.53	78.22$\pm$0.71	1.53	66
+DynaFed [29]	90.02	91.44	95.04	89.97	91.62$\pm$0.45	-0.32	-	82.30	64.44	72.22	80.51	74.87$\pm$1.67	-1.82	-
+FedFTG [19]	87.23	93.95	91.52	89.21	90.48$\pm$0.34	-1.46	-	81.25	64.89	74.26	81.27	75.42$\pm$1.23	-1.27	-
+FGL [20]	90.43	89.53	94.94	89.47	91.09$\pm$0.31	-0.85	-	68.77	63.78	85.19	86.27	76.00$\pm$0.15	-0.69	-
+Flick	91.36	94.27	95.83	93.46	**93.73**$\pm$0.16	**1.79**	**16**	80.21	65.33	85.19	94.92	**81.41**$\pm$0.15	**4.72**	**16**
FedDyn [43]	83.94	91.93	92.11	91.75	89.93$\pm$0.44	-	22	79.17	62.44	83.33	88.14	78.27$\pm$0.12	-	68
+FedBN [28]	86.78	91.30	92.66	91.50	90.56$\pm$0.50	0.63	22	81.64	65.44	84.26	90.68	80.51$\pm$0.43	2.24	79
+FedHEAL [27]	87.59	92.57	94.94	91.56	91.67$\pm$0.10	1.74	**16**	73.44	61.11	87.04	94.92	79.13$\pm$0.26	0.86	73
+DynaFed [29]	87.75	91.86	94.25	90.61	91.12$\pm$0.29	1.19	20	79.67	64.67	76.89	91.11	78.09$\pm$0.71	-0.18	-
+FedFTG [19]	86.62	93.10	93.45	91.62	91.20$\pm$0.47	1.27	20	80.73	65.44	80.18	84.14	77.62$\pm$0.69	-0.65	-
+FGL [20]	88.69	93.63	92.86	92.01	91.79$\pm$0.06	1.86	**13**	82.29	67.85	82.72	93.22	81.52$\pm$0.76	3.25	89
+Flick	90.02	95.01	95.39	93.15	**93.39**$\pm$0.33	**3.46**	16	84.11	65.56	87.03	94.07	**82.69**$\pm$0.59	**4.42**	**23**
FedNAR [16]	89.86	92.92	96.03	89.81	92.16$\pm$0.13	-	84	77.99	63.00	75.93	92.80	77.43$\pm$0.86	-	66
+FedBN [28]	90.75	92.57	94.94	90.61	92.22$\pm$0.34	0.06	142	80.90	64.00	76.54	89.27	77.68$\pm$0.70	0.25	131
+FedHEAL [27]	91.24	95.12	95.24	90.61	93.05$\pm$0.47	0.89	30	80.21	63.56	77.78	91.53	78.27$\pm$0.64	0.84	59
+DynaFed [29]	89.90	92.94	94.72	89.88	91.86$\pm$0.39	-0.30	-	81.52	67.56	74.04	88.81	77.98$\pm$0.72	0.55	101
+FedFTG [19]	88.81	93.84	91.96	89.72	91.08$\pm$0.72	-1.08	-	82.73	65.11	73.81	81.34	75.75$\pm$1.06	-1.68	-
+FGL [20]	90.39	90.76	95.09	89.78	91.51$\pm$0.12	-0.65	-	72.99	64.33	83.33	84.34	76.25$\pm$0.10	-1.18	-
+Flick	92.34	93.74	95.98	92.70	**93.69**$\pm$0.18	**1.53**	**16**	80.38	65.33	88.89	88.70	**80.83**$\pm$0.54	**3.40**	**25**

space but only from one domain. It shows that Flick consistently yields significant improvement in model accuracy and reduces the number of communication rounds (#Round) to reach the target accuracy–defined as the best accuracy of the baseline methods (i.e., FedAvg, FedProx, FedDyn, FedNAR). Among generation-based counterparts, DynaFed and FedFTG have limited performance gain under the heterogeneous settings with domain shift alone and even show inferior performance to the baselines on the Office-Caltech dataset. For FGL, we match the synthetic data volume to that of Flick. For example, under the FedAvg baseline on the PACS dataset, Flick generates only 300 synthetic samples in total. This limited amount of auxiliary data constrains the performance of FGL – a finding consistent with its released paper. In contrast, the same amount proves sufficient for Flick to deliver significant performance improvements. More importantly, compared to methods specifically designed to address domain shift (i.e., FedBN and FedHEAL), Flick further improves the global model performance and convergence by a large margin. Combined with the results from Table 1, it is clear that Flick shows great performance in addressing data heterogeneity issues regarding label skew and domain shift.

C.4 Scalability Analysis on DomainNet Dataset

To evaluate the scalability of Flick in large-scale federated settings, we experiment on the DomainNet dataset [51] with 100 clients, where 20% are randomly selected to participate in each communication round. The DomainNet dataset consists of natural images with six different domains: Clipart, Infograph, Painting, Quickdraw, Real, and Sketch. Following prior work [20, 28, 52], we construct a sub-dataset for our experiments by selecting the top ten most common classes: airplane, clock, axe, basketball, bicycle, bird, strawberry, flower, pizza, bracelet. We partition each domain into 15 or 17 subsets using a Dirichlet distribution with concentration parameter $\alpha = 0.1$. Each client receives data from a single domain with a skewed label distribution, simulating a scenario with both label skew and domain shift.

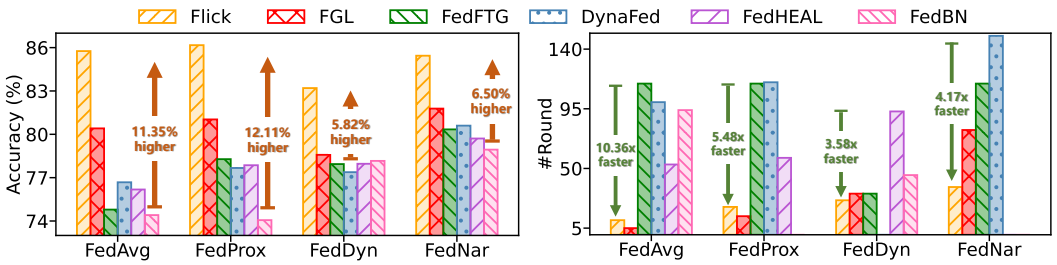
Table 8 reports the model performance across various baselines. Flick consistently achieves superior performance in both Top-1 accuracy and round-to-accuracy (#Round), demonstrating its robustness in large-scale heterogeneous FL. Specifically, Flick improves model accuracy by up to 11.63% and

Table 8: Global model accuracy performance (%) under both domain shift and label skew on DomainNet dataset. AVG denotes the average accuracy calculated across all domains, and Δ indicates the accuracy gain compared with vanilla methods. **Best** in bold.

Methods	DomainNet								
	Clipart	Infograph	Painting	Quickdraw	Real	Sketch	AVG	Δ ↑	#Round ↓
FedAvg [5]	82.59	46.99	71.60	77.63	85.95	80.00	74.13±0.38	-	144
+FedBN [28]	83.64	52.00	72.41	69.78	91.30	77.36	74.41±0.17	0.28	94
+FedHEAL [27]	86.50	53.18	74.33	72.42	90.65	80.00	76.18±0.30	2.05	53
+DynaFed [29]	86.13	50.31	69.22	85.90	87.16	81.40	76.68±1.27	2.55	100
+FedFTG [19]	84.31	50.00	71.20	80.87	87.69	74.69	74.79±0.32	0.66	114
+FGL [20]	91.09	55.56	79.71	79.07	92.21	84.80	80.41±0.39	6.28	**5**
+Flick	92.11	63.90	83.88	91.40	92.91	90.34	**85.76**±0.07	**11.63**	11
FedProx [9]	85.32	52.59	72.99	83.37	88.15	80.76	77.20±0.60	-	144
+FedBN [28]	83.48	52.73	72.34	66.63	91.49	77.66	74.06±0.02	-3.14	-
+FedHEAL [27]	87.36	54.84	74.88	75.91	92.33	81.86	77.86±0.12	0.66	58
+DynaFed [29]	86.94	53.94	73.18	82.60	87.56	81.79	77.67±0.11	0.47	115
+FedFTG [19]	86.82	51.79	73.86	86.07	89.72	81.45	78.28±0.30	1.08	114
+FGL [20]	91.40	54.98	80.98	81.00	92.79	85.03	81.03±0.48	3.83	**14**
+Flick	92.81	64.94	84.94	91.60	93.07	89.66	**86.17**±0.05	**8.97**	21
FedDyn [43]	88.06	55.39	77.08	75.60	90.32	81.03	77.91±0.22	-	34
+FedBN [28]	87.22	54.79	78.82	77.90	89.72	80.48	78.16±0.33	0.25	45
+FedHEAL [27]	86.51	56.97	74.66	75.03	91.89	82.68	77.96±0.21	0.05	93
+DynaFed [29]	84.41	53.53	74.11	81.73	89.46	81.03	77.38±0.21	-0.53	-
+FedFTG [19]	86.13	54.05	74.50	82.20	88.80	81.93	77.94±0.59	0.03	31
+FGL [20]	87.35	54.67	76.55	80.17	89.99	82.76	78.58±0.13	0.67	31
+Flick	89.68	60.37	83.09	87.87	91.14	87.03	**83.20**±0.13	**5.29**	**26**
FedNAR [16]	89.88	56.85	75.36	86.10	91.69	81.93	80.30±0.40	-	150
+FedBN [28]	85.50	58.22	76.50	78.90	93.33	81.17	78.94±0.14	-1.36	-
+FedHEAL [27]	88.43	57.05	77.37	77.87	93.87	83.69	79.71±0.02	-0.59	-
+DynaFed [29]	89.77	56.08	77.16	83.63	92.60	84.38	80.60±0.21	0.30	150
+FedFTG [19]	90.72	53.10	74.63	84.50	94.71	84.41	80.34±0.10	0.03	114
+FGL [20]	90.38	57.47	81.11	83.83	92.65	85.24	81.78±0.07	1.48	79
+Flick	91.30	64.73	84.02	90.40	93.11	89.10	**85.44**±0.17	**5.14**	**36**

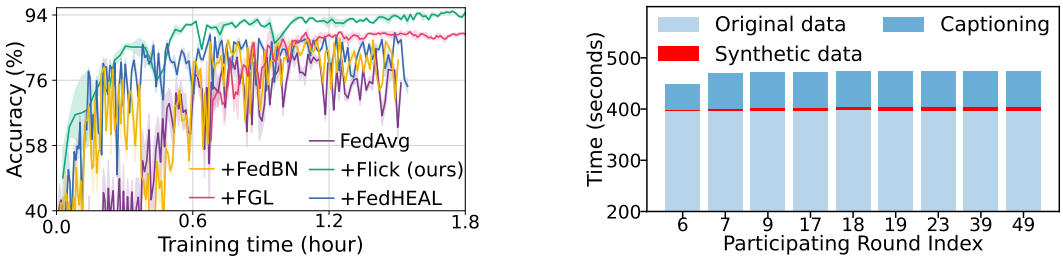
reduces the number of rounds required to reach the target accuracy from 144 to 11. Although FGL performs fast convergence on FedAvg and FedProx baseline methods by fine-tuning the global model on a large IID synthetic dataset, the significant overhead required for data generation limits its scalability in practical federated settings.

We fix the synthetic data volume for FGL at 1500, while the total number of generated samples (#Total Sample) in Flick is lower – 1457, 1133, 1412, and 1299 for FedAvg, FedProx, FedDyn, and FedNova, respectively. Besides, the number of samples required to reach the target accuracy is only 427, 414, 417, and 477, highlighting that Flick achieves superior performance with substantially less generative overhead. Figure 13 shows that Flick outperforms all counterparts in model accuracy by up to 11.35%, 12.11%, 5.82%, and 6.50%, and accelerates convergence by up to 10.36×, 5.48×, 3.58×, and 4.17× across four baseline methods.



(a) Model accuracy. (b) Round-to-accuracy.

Figure 13: Model performance across various baseline methods on the DomainNet dataset.



(a) Time-to-Accuracy performance

(b) Local latency over rounds

Figure 14: Evaluation on the temporal scale: **(a)** Wall-clock training time of Flick and its model-driven counterparts on the PACS dataset, running on an NVIDIA A40 GPU; **(b)** Wall-clock local latency of a specific client, running on an NVIDIA Jetson AGX Orin, across participating rounds, including training on both original and synthetic data points, as well as local captioning.

D Overheads

Compared to the model-driven methods for heterogeneous FL, the data-driven Flick introduces additional overheads to the clients, including extra training and captioning latency. However, the performance gain from Flick remains significant: it facilitates the model convergence, as reflected in both reduced round-to-accuracy (#Round) and training time, while also improving the top-1 accuracy of the global model in the long run. Figure 14(a) shows the learning curves of Flick and counterparts, integrated into the baseline method FedAvg, with time-to-accuracy performance evaluated on an NVIDIA A40 GPU. The faster convergence offered by Flick significantly counteracts the negative impact from additional overheads. Besides, we also trace local latency variation over communication rounds by measuring the time consumption of a single client running on an NVIDIA Jetson AGX Orin, as shown in Figure 14(b). We break down the additional overheads into training on synthetic data and captioning the representative samples, which employs the “Salesforce/blip-image-captioning-large" with ViT and BERT as image and text encoders. It is obvious that the local overheads of Flick stabilize over time and constitute only a small fraction of the overall local latency as training proceeds. In contrast, FGL – the best-performing counterpart – incurs substantial delay in the first communication round due to the extensive preparation of synthetic data, leading to a longer time to reach the target accuracy.

E Privacy Analysis

Flick can ensure the ϵ-DP guarantee by adding noise sampled from the Laplace distribution to the token sequences, with the scale parameter in the Laplace mechanism set to $1/\epsilon$. It provides stronger privacy protection as ϵ decreases. Table 9 shows the trade-off between system utility (i.e., Flick performance) and privacy level, with the ϵ ranging from 0.05 to 0.5. Note that the performance gradually degrades towards the Flick *w/o Local Summary* variant as the ϵ decreases. Even under strict privacy constraints, Flick maintains promising model performance.

Table 9: The performance of Flick across different ϵ values.

ϵ	PACS			Office-Caltech		
	AVG ↑	Δ ↑	#Round ↓	AVG ↑	Δ ↑	#Round ↓
$\epsilon = 0.5$	94.32±0.21	11.26	11	78.24±0.15	7.40	38
$\epsilon = 0.1$	93.78±0.24	10.72	25	77.61±0.33	6.77	43
$\epsilon = 0.05$	92.99±0.31	9.93	31	77.05±0.32	6.21	43

Besides, we assess the potential information leakage in local summaries from a different angle: the difficulty for the server to infer raw data from collected token sequences. To this end, the central server feeds the local summaries (in the textual format) to a text-to-image model, and we evaluate five different models. Table 10 reports the reconstruction quality based on three widely used metrics–peak signal-to-noise ratio (PSNR) [54], structural similarity index measure (SSIM) [55], and learned perceptual image patch similarity (LPIPS) [56]–which measure the similarity between reconstructed

Table 10: Privacy assessment of local summary phase in Flick. The server utilizes five image generation models to reconstruct the data from collected local summaries. The similarity between original raw local data and reconstructed data is measured by three widely used metrics, with reference ranges provided to indicate the threshold below/beyond which pair-wise images can be considered dissimilar. *PSNR [54]: peak signal-to-noise ratio; SSIM [55]: structural similarity index measure; LPIPS [56]: learned perceptual image patch similarity.*

Metrics	dall-e-2	dall-e-3	v1.5	v1.4	xl-base-1
PSNR (< 20)	7.77	8.11	7.79	7.80	9.37
SSIM (< 0.3)	0.20	0.16	0.17	0.14	0.20
LPIPS (> 0.6)	0.72	0.74	0.78	0.77	0.72

images generated by server-held models and the original raw local images. For PSNR and SSIM, higher values indicate greater similarity, whereas for LPIPS, lower values indicate higher similarity. The results across all five image generators show that reconstructed images remain significantly dissimilar to the original raw data across three metrics, demonstrating that uploading local summaries in Flick presents a low privacy risk.

F Data Generation Details

In Section 3.3, we have introduced the data generation process in Flick. In this section, we provide some more details – specifically for the LLM usage and data retrieval – by showing the intermediate results.

The data generation phase bootstraps the server to obtain the class-wise decision matrix $\mathbf{D}$. After that, an LLM is employed to generate text prompts for each class. Figure 15 provides a detailed case example to supplement Figure 3 in the paper, illustrating how the LLM produces $G = 5$ text prompts for the class *dog*, using a collection of local captions $\mathcal{T}$ as reference. Recall that, for privacy considerations, our design in the local summary phase only requires participating clients to report captions of representative local samples while omitting class information. Consequently, the LLM has to first identify and select class-related captions from the pool $\mathcal{T}$. From the LLM output in Figure 15, it is evident that the LLM can effectively select captions relevant to the class *dog*, extending beyond simple keyword matching. For instance, it also identifies captions containing terms like

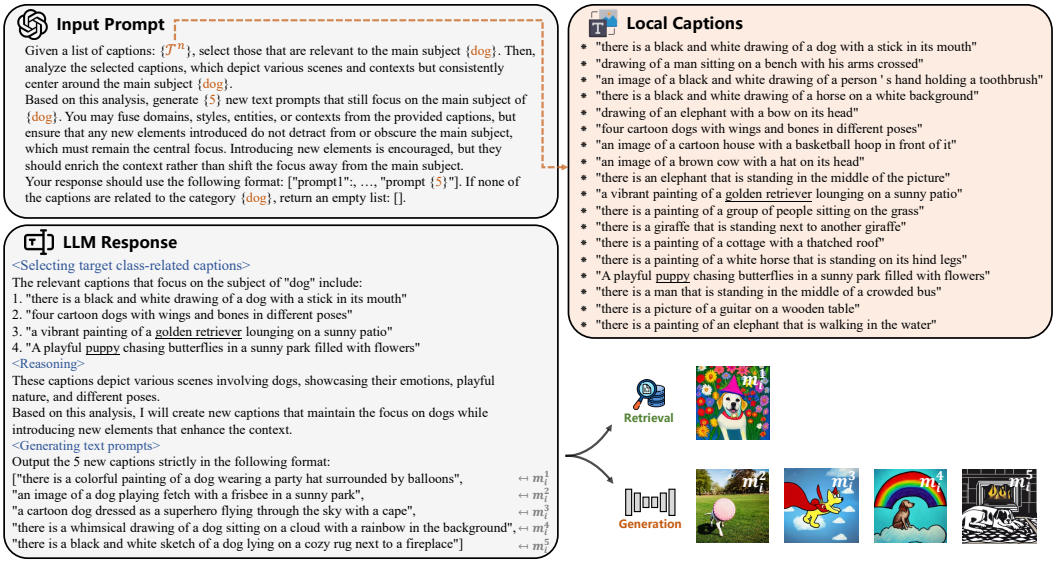


Figure 15: Detailed illustration of LLM-based text prompt generation: given cross-client-specific knowledge $\mathcal{T}^n$, the central server obtains the text prompts for the class *dog* in the PACS dataset. The underlined words are recognized by LLM as related to class *dog*.

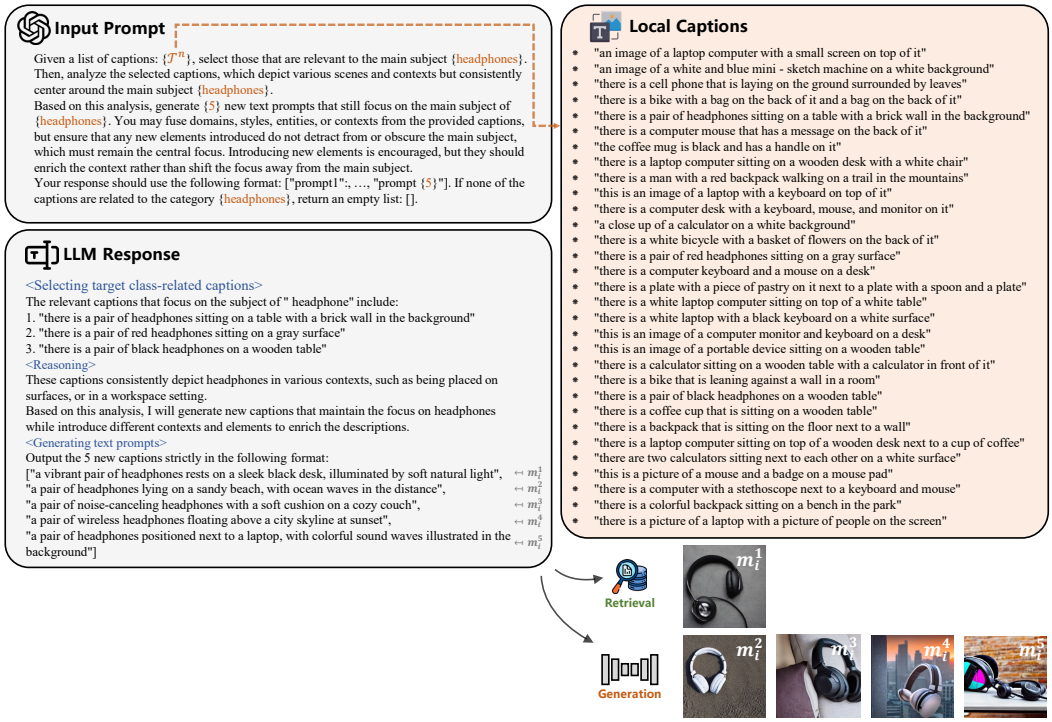


Figure 16: Detailed illustration of the LLM-based text prompt generation for the class *headphones* in the Office-Caltech dataset.

"golden retriever" or "puppy". During the reasoning phase, the LLM analyzes the selected captions and recombines extracted features (i.e., cross-client-specific knowledge) with new elements (i.e., commonsense knowledge embedded in the LLM) to generate text prompts for the subsequent text-to-image model. Besides, Figure 16 illustrates the text prompts generation for the class *headphones* in the Office-Caltech benchmark where the LLM showcases similar capabilities in extracting class-related knowledge and instilling commonsense knowledge.

To reduce the data generation overheads, the server in our Flick does not directly feed all text prompts into a generative model for new data points. Instead, it first retrieves historically generated samples from the server-held data pool $\mathcal{G}_s$. The qualified historical samples are retrieved based on the pairwise similarity between the historical text prompt $m_s^p \in \mathcal{M}_s$ and the generated text prompt $m_i^q \in \mathcal{M}_i$. Specifically, each prompt pair (m_s^p, m_i^q) is transformed into SBERT embeddings used for calculating cosine similarity. In Figure 17, we provide more examples of the data retrieval step across two benchmarks and their respective four domains. For detailed illustrations, Figure 17 presents the text prompt pairs (m_s^p, m_i^q) and their SBERT similarity scores, as well as the sample corresponding to the historical text prompt m_s^p that the server retrieves.

Figure 17: Illustration of the retrieved samples in two datasets: PACS (*top*) and Office-Caltech (*bottom*). In each example, m_s^p and m_i^q represent historical text prompt and generated text prompt, respectively. In Flick, the server retrieves samples from the data pool for which the corresponding text prompts show high similarity to the generated text prompts.