Pattern Recognition Letters 197 (2025) 195–201

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Benchmarking federated learning for semantic datasets: Federated scene graph generation

SeungBum Ha [a,1], Taehwan Lee [a,1], Jiyoun Lim [c], Sung Whan Yoon [a,b,*]

[a] Graduate School of Artificial Intelligence, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea
[b] Department of Electrical Engineering, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea
[c] Electronics and Telecommunications Research Institute (ETRI), Daejeon, Republic of Korea

ARTICLE INFO

Editor: Jiwen Lu

Keywords:
Scene graph generation
Panoptic scene graph generation
Federated learning
Distributed learning
Data privacy
Benchmark

ABSTRACT

Federated learning (FL) enables decentralized training while preserving data privacy, yet existing FL benchmarks address relatively simple classification tasks, where each sample is annotated with a one-hot label. However, little attention has been paid to demonstrating an FL benchmark that handles complicated semantics, where each sample encompasses diverse semantic information, such as relations between objects. Because the existing benchmarks are designed to distribute data in a narrow view of a single semantic, managing the complicated *semantic heterogeneity* across clients when formalizing FL benchmarks is non-trivial. In this paper, we propose a benchmark process to establish an FL benchmark with controllable semantic heterogeneity across clients: two key steps are (i) data clustering with semantics and (ii) data distributing via controllable semantic heterogeneity across clients. As a proof of concept, we construct a federated PSG benchmark, demonstrating the efficacy of the existing PSG methods in an FL setting with controllable semantic heterogeneity of scene graphs. We also present the effectiveness of our benchmark by applying robust federated learning algorithms to data heterogeneity to show increased performance. To our knowledge, this is the first benchmark framework that enables federated learning and its evaluation for multi-semantic vision tasks under the controlled semantic heterogeneity. Our code is available at https://github.com/Seung-B/FL-PSG.

1. Introduction

Federated learning (FL) has drawn considerable attention as a key framework to enable decentralized deep model training from private data of numerous clients. The FL framework communicates the model parameters between the clients and the server; to keep the distributed local data private, the server cannot access data samples of clients [1]. The property that FL preserves data privacy makes it more crucial when deep models handle license- or privacy-sensitive data, e.g., clinical data from medical institutions, licensed content from providers, and broadcasting stations.

Along with the rapid algorithmic development of FL, significant efforts have been dedicated to constructing FL benchmarks that enable reliable and rigorous evaluations of methods. The existing FL benchmarks mostly rely on the existing datasets, such as CIFAR [2] and Twitter [3], etc. Therefore researchers focus on devising a decentralized training setting with controllable factors, such as data heterogeneity across clients, number of clients, and participation ratio. Among the factors of FL settings, *data heterogeneity* works as the most crucial factor that exhibits the efficacy of different FL algorithms; when the data distribution strongly deviates across clients, a federation of local models typically fail with drastic performance drops [4]. Researchers have mainly focused on data heterogeneity constructed by a single label, which is straightforward. Therefore, prior works diversify the distribution across clients via Dirichlet distribution [5] or shard- or chunk-wise assignment of data [1].

Herein, we point out two key limitations of the existing FL benchmarks. Firstly, the current benchmarks mainly handle classification or regression tasks, where each sample consists of a single label. However, deep learning tasks are becoming far beyond classification or recognition, and complicated jobs are being considered to understand in-depth semantic information. Therefore, extending the current FL benchmark process to complicated semantics is necessary.

Second, there does not exist a task-agnostic FL benchmark process that devises controllable *semantic heterogeneity*. The classification task of FL focuses on a single semantic (the class label) in the left of Fig. 1 and constructs heterogeneity based on the label. In contrast, devising

* Corresponding author at: Graduate School of Artificial Intelligence, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea.
E-mail addresses: ethereal0507@unist.ac.kr (S. Ha), taehwan@unist.ac.kr (T. Lee), kusses@etri.re.kr (J. Lim), shyoon8@unist.ac.kr (S.W. Yoon).
[1] Co-first author

https://doi.org/10.1016/j.patrec.2025.07.020
Received 26 March 2025; Received in revised form 9 June 2025; Accepted 23 July 2025
Available online 6 August 2025

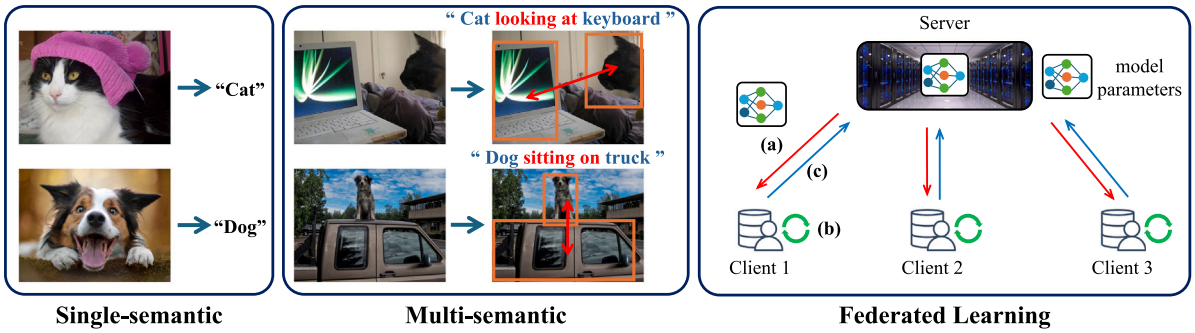

Fig. 1. Examples of single-semantic (Left: classification) and multi-semantic (Middle: scene graph generation) tasks, and the overview of the federated learning process (Right).

heterogeneity is non-trivial when each sample contains multiple semantics. As shown in the middle of Fig. 1, Scene Graph Generation, which understands the complicated semantics of an image, a single image contains multiple objects (‘cat’, ‘keyboard’, ‘dog’ and ‘truck’), predicates (‘looking at’ and ‘sitting on’) and relations (‘cat’ → ‘keyboard’, ‘dog’ → ‘truck’). It remains unexplored to construct the heterogeneity with controllable and complex semantics.

In this study, we propose a comprehensive FL benchmark process for evaluating FL algorithms on multi-semantic datasets while controlling semantic heterogeneity. To break the limitations of existing studies, our process encompasses two key steps: (i) discovering the semantic clusters by utilizing the collection of multiple annotations. and (ii) distributing data samples to multiple clients by considering the heterogeneity. The simulation results reveal that the methods tailored to tackle the long-tailed problem in the Panoptic Scene Graph Generation (PSG) task, where some objects and predicates are more dominant than others, are robust in handling semantic heterogeneity in FL.

2. Related works

2.1. Federated learning (FL) and benchmarks

FL has emerged as a framework for training deep learning models in a decentralized setting, enabling the preservation of data privacy for clients. We briefly introduce preliminaries by focusing on the foundational baseline, i.e., FedAvg, [1]. FL setting contains the single server and K clients. The training process proceeds iteratively in rounds. Each round follows the three-step procedure shown on the right of Fig. 1, summarized as follows: **(a)**: The server distributes a global model to clients. **(b)**: Each client initializes the local model with the distributed one and trains a neural network using their dataset **(c)**: Clients upload the locally trained model parameters to the server. Then, the server averages the aggregated models and performs the step **(a)** again. The aggregation step is represented as $w^{t+1} = \sum_{k=1}^{S_t} \frac{n_k}{n} w_k^t$, where n_k is the number of data samples on client k, S_t is the set of selected clients at round t, and n is the total number of data samples across S_t clients. When S_t equals K, all the clients participate in the aggregation step, and in the case of $S_t < K$, partial clients participate in the aggregation step.

In FL systems, each user has independent data with different distributions, which can cause the trained local models to diverge, negatively impacting the performance and convergence of FL. Therefore, researchers have mainly been dedicated to handling the case with a substantial heterogeneity of data across clients, resulting in diverse strategies. For example, FedAvgM [6] leverages global-model updates as momentum. At each round, the server keeps a momentum vector formed by accumulating the difference between consecutive global models. FedAdam [7] treats update differences as gradients optimized by Adam. Recently, regularization-based methods such as ℓ_1-Fed [8], which alleviates similar distribution through the addition of an ℓ_1 sparsity constraint to the global model, have also been reported. And FedGF [9], which resolves the difference between the local and global objectives by raising the generalization ability with Sharpness-Aware Minimization [10].

When we construct a heterogeneous distribution with a simple semantics, such as a single target label, unified strategies exist to impose heterogeneity across clients by diversifying the prior distribution of the target label [11]. Specifically, two main strategies include **(i)** sampling the prior distribution of each client from Dirichlet distribution [5], and **(ii)** chunking per-class data samples into multiple shards, where a fixed number of shards is allocated to each client, resulting in heterogeneity between clients [12,13].

2.2. Panoptic scene graph generation (PSG)

Scene graphs are crucial for scene understanding in computer vision tasks, representing objects (nodes), denoted by bounding boxes or pixel-wise segmentation, and predicates (relationships, edges) in a graph structure [14–16]. Predicting the bounding boxes and relationships between bounding boxes constitutes scene graph generation. The PSG task has been proposed by [17,18], delving deeper into scene graph generation using panoptic segmentation masks instead of bounding boxes. The difference between PSG and classic scene graph generation is that PSG uses panoptic segmentation [19] masks rather than bounding boxes. Moreover, the scene graph generation tasks face the long-tailed problem [20,21]. Positional relationships among objects constitute the majority of the predicates, leading to a visual relationship complexity of $\mathcal{O}(N^2R)$ for N objects and R predicates [22]. This exacerbates the long-tailed problem in SGG datasets, prompting various approaches, such as utilizing self-attention to characterize complex interactions, thereby facilitating the understanding of object and relation semantics [23], and estimating confidence scores and weighting high-uncertainty cases more heavily during training [24]. Also, the self-supervised local pseudo-attribute is utilized to reinforce tail-class representations [25].

To perform challenging vision tasks such as PSG, having more data typically leads to training better models. However, in reality, the photos held by different clients are unlikely to be similar, and collecting such data on the server for training poses a threat to data privacy. Despite this, no attempts have been made to apply FL to the PSG task, underscoring the necessity of this research. Moreover, the data heterogeneity issue in FL closely resembles the well-known long-tailed problem in scene graph generation tasks and real-world data [21].

3. A benchmark process for FL with multi-semantic datasets

For a given multi-semantic dataset, each data sample contains multiple annotations, i.e., $(x, \mathcal{Y}) \in D$, where x is an input, $\mathcal{Y} = \{y_1, \ldots, y_M\}$ is a set of multi-semantic labels, M is the possible number of labels for each data sample, and D represents the dataset. Here, we introduce our benchmark process to distribute the multi-semantic data samples to clients with controllable semantic heterogeneity. The key steps are

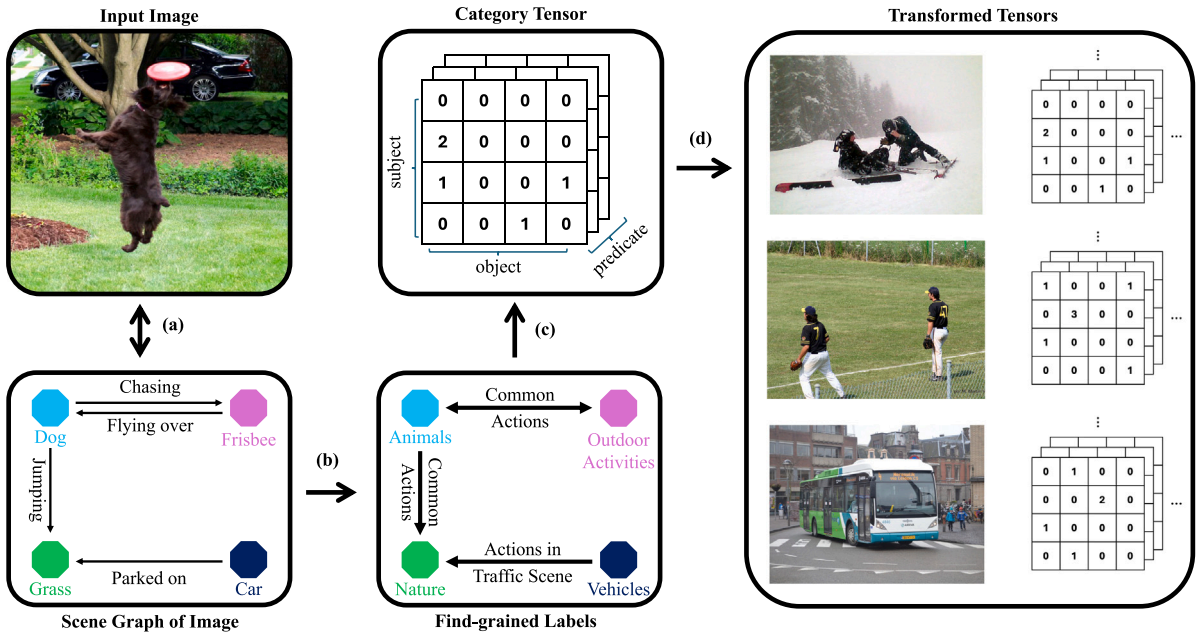

Fig. 2. Category tensor K-means clustering pipeline. **(a)** Construct a graph using the objects, subjects, and predicates of each image. **(b)** Map each label into the super-classes of fine-grained labels. **(c)** Convert categorized relations (subject, object, and predicate) into the category tensor. **(d)** Transform every input image into a category tensor and perform K-Means Clustering.

twofold: (i) discovering data clusters with different semantics and (ii) data partitioning with controllable semantic heterogeneity across clients.

3.1. Discovering data clusters: K-means clustering of category tensor

For a given multi-semantic $\mathcal{Y}$, we transform it to *category tensor* $\mathcal{T}$ by allocating each label y_i into an orthogonal axis of the tensor, i.e., $\mathcal{T}(\mathcal{Y}) \in \mathbb{R}^{N_1 \times \cdots \times N_L}$, where there are $N_1, \ldots, N_L$ possible categories for each respective label of $\mathcal{Y}$. We then apply K-means Clustering on the collection of $\mathcal{T}(\mathcal{Y})_1^{|\mathcal{D}|}$ of overall dataset:

$$\mathcal{K}\left(\mathcal{T}(\mathcal{Y})_1^{|\mathcal{D}|}\right) \rightarrow \{C_1, \ldots C_n\}, \tag{1}$$

where n is the number of clusters that can be determined depending on the dataset and C_i indicates the collection of samples assigned to ith cluster. With the obtained clustering, we can transform each data sample $(x, \mathcal{Y})$ into (x, C_i) to impose the cluster label with semantic information, which is a one-hot label with 1 for the assigned cluster. As a result of the clustering process, we can perform the label-based partition while fully utilizing the multi-semantic information of each data sample with its corresponding cluster label C. We present the overall pipeline to Fig. 2.

3.2. Data partition with semantic heterogeneity

We acquire n clusters from Eq. (1). It trivially raises the issue that the clusters are not evenly distributed, so the number of samples assigned to each cluster would deviate for different clusters, i.e., *cluster imbalance*. The *cluster imbalance* prevents rigorous evaluations of FL models to handle semantic heterogeneity because a model becomes overfitted to dominant clusters without balanced training across different semantics. The cluster imbalance stems from the long-tailed problem, a key challenge in scene graph generation datasets. In other words, we have to create data heterogeneity for FL, which further complicates distinguishing it from the long-tailed problem. If the amount of data in each cluster is equalized, the long-tailed problem can be effectively alleviated. Furthermore, considering the FL scenario, this cluster imbalance will likely bias the update of the global model in the direction of users belonging to the dominant cluster. It causes overfitting to a dominant cluster, which makes it difficult to fairly compare each method closely. Consequently, we need to equalize the data quantity of each cluster: $\hat{C}_k = \text{Sample}(C_k, m)$, for all $1 \leq k \leq n$, where $m = \min_{k \in [n]} \{|C_k|\}$, $|C_k|$ is the cardinality of the kth cluster C_k, and $\text{Sample}(C_k, m)$ functions to randomly select m data samples from cluster C_k. We apply the label-based partition based on these clusters to impose semantic heterogeneity. Our benchmark suggests two partition strategies as follows.

Shard-based partition: Each client chooses $p(\leq n)$ clusters. We then split each cluster into disjoint shards or chunks, where the number of shards equals the number of clients who selected the cluster. After splitting, the shards are distributed to the corresponding clients. If $p = n$, all clients are assigned to all clusters, making the data distribution homogeneous. The distribution is close to heterogeneous for smaller p values.

Dirichlet distribution-based partition: From the strategy suggested in [5], the amount of data each client takes from cluster k is governed by the sampling from the Dirichlet distribution. We design the non-IID data partition into U clients by sampling a multinomial probability vector for each client u, denoted as: $\mathbf{p}_u \sim \text{Dir}_n(\boldsymbol{\alpha})$ where the probability vector is $\mathbf{p}_u = (p_{u,1}, p_{u,2}, \ldots, p_{u,n})$, and the concentration parameter of Dirichlet distribution is $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, for $\alpha_i > 0, \forall i \in \{1, 2, \ldots, n\}$. Each uth client samples training data from a dataset according to the proportion $p_{u,i}$ without replacement for each cluster i. The data heterogeneity is controlled by $\boldsymbol{\alpha}$: As $\boldsymbol{\alpha} \rightarrow \infty$, the data distribution goes to IID; in contrast, as $\alpha \rightarrow 0$, it goes to non-IID.

3.3. Proof-of-concept: FL benchmark for panoptic scene graph generation (PSG)

We provide a proof-of-concept of our FL benchmark process by constructing the FL benchmark for PSG dataset.

(i) Discovering data clusters: PSG dataset contains object, subject, and predicate labels for each image sample. For simplicity, we utilize 13 object/subject categories and 7 predicate categories, which are the super-classes of fine-grained labels Therefore, the dimension of the category tensor is $\mathcal{T}(\mathcal{Y}) \in \mathbb{R}^{13 \times 13 \times 7}$. We perform K-means Clustering

for the category tensor to obtain multiple semantic clusters, obtaining 5 clusters with discriminated semantics.[2]:

$$\mathcal{K}\left(\mathcal{F}(\mathcal{Y})_1^{|\mathcal{D}|}\right) \rightarrow \{C_1, C_2, C_3, C_4, C_5\}. \quad (2)$$

(ii) Data partition: Based on the discovered 5 semantic clusters, our benchmark provides two options for data distribution: (i) Shard-based partitioning and (ii) Dirichlet distribution-based partitioning. As partitioning becomes heterogeneous, the data distribution at clients strongly deviates in the sense of semantic clusters. Otherwise, the data distribution of clients becomes homogeneous, yielding evenly distributed semantic information.

4. Experiments: Benchmarks for PSG in FL

4.1. Experiment settings

We evaluate the existing panoptic scene graph generation (PSG) models on our benchmark with the following methods: IMP [26], MOTIFS [27], VCTree [28], and GPS-Net [29]. Because these PSG methods use the same pretrained object detector Faster R-CNN [30], and the communication cost is crucial in FL scenarios, we freeze the pretrained object detector and focus on predicate classification. Therefore, each client trains and aggregates the relation head, responsible for processing predicates by capturing the semantic relationships between objects.

An imbalanced dataset contains a number of data points in each cluster that is not equal after clustering. We randomly sampled the data from each cluster to match the quantity of the smallest cluster to eliminate cluster imbalance. This process ensured that all clusters had the same data (2.2K images), resulting in a balanced dataset.

Experiment setups: We set up an FL scenario with one server and 100 clients, distributing the training data of the existing PSG dataset [17] to the 100 clients. The test data for our benchmark is the same as the PSG test dataset. Five active clients are randomly selected in each round, and the test data is evaluated using the aggregated global model from the server. Each client performs local training with one epoch and a batch size 16. The total number of training rounds is 100, and we report the R/mR@K performance of the final averaged model. Following the benchmark in [17], we set the SGD optimizer to a local optimizer with a learning rate of 0.02, momentum of 0.9, weight decay of 0.0001, and gradient clipping with a max L2 norm of 35.

Clustered PSG Dataset Description: By examining the samples for each cluster, we observe the following features for each cluster and the imbalance between clusters[3]:

- **Cluster 1** (occupying 5% of datasets)
  We observe that it contains a large number of **animal objects** compared to others. The predicates are composed of actions that animals trivially perform.
- **Cluster 2** (occupying 58% of datasets)
  This cluster is dominated by **daily photographs of people**, which constitutes the largest portion of PSG dataset. This cluster is mainly related to daily activities by human beings that frequently appear in daily life.
- **Cluster 3** (occupying 11% of datasets)
  This cluster mainly includes urban landscape and transportation photos, which encompass many predicates related to vehicles, such as 'parking on' and 'driving (on).'
- **Cluster 4** (occupying 7% of datasets)
  This cluster is composed of **sports** and **kinetic** images, containing predicates such as 'playing,' which are more prevalent than others.
- **Cluster 5** (occupying 19% of datasets)
  This cluster corresponds to **urban/nature-combined landscapes**, which typically include buildings, the sky, and a river in the images. Due to objects related to natural elements, the predicates in this cluster are predominantly positional rather than action-oriented.

Notably, Clusters 2 and 4 contain somewhat similar images, mainly of 'people'. However, the predicates in Cluster 4 relate to sports, clearly distinguishing it from Cluster 2. Also, Cluster 3 and 5 look similar because of urban landscapes, but Cluster 3 tends to focus on cityscapes with transportation, and Cluster 5 focuses on urban/nature-combined views.

Benchmark setups: We randomly sampled data from each cluster to ensure an equal amount of data for each cluster to ease the cluster imbalance. We test 6 types of data partitioning as follows:

(1) Random: Data is distributed randomly among all clients, ensuring nearly equal sizes for each.

(2) Shard-based partition IID: We set $p = 5$, where p is the number of clusters that client sample from. When p equals the number of clusters, the data from each cluster is equally distributed among 100 clients.

(3) Shard-based partition non-IID: We set $p = 1$ for imposing semantic heterogeneity. Each cluster is assigned 20 clients, and all clients have the same amount of data.

(4), (5) and (6) Dirichlet distribution-based partition: We test three levels of semantic heterogeneity by using $\alpha = [10, 1, 0.2]$ to simulate from an IID to a non-IID case.

Metrics: By following [17] which suggested the PSG task, we use 'Recall@K (R@K)' and 'mean Recall@K (mR@K)' as the performance metrics, which calculate the triplet recall and mean recall for every predicate category, given the top $K \in [20, 100]$ triplets from a PSG method. Moreover, R@K is dominated by high-frequency relations, and mR@K assigns equal weight to all relation classes. In datasets with severe long-tailed problems, e.g., PSG dataset, mR@K can provide more meaningful insights into model performance.

4.2. In-depth analysis

Our intuition is that the performance of models is expected to show the following order: Centralized learning(CL) c≥ IID ≥ Random ≥ non-IID, when our benchmark effectively imposes semantic heterogeneity in the FL setting. The experimental results also follow our intuition and validate the effectiveness of our benchmark.

Results: Table 1 shows the test accuracy on the test set of the PSG dataset.[4] We have focused on the Mean Recall ('mR') performance. Also, we focus on the most challenging case with $K = 20$.

(i) CL vs. IID. The performance has been mostly degraded when comparing CL and IID cases. The averaged gaps for mR@20 are −2.45% and −2.71%, for 'Shard-IID' and 'Dir($\alpha = 10$)'. Each client has approximately 114 images, and due to the limited data, there appears to be a performance difference between the CL and IID scenarios. CL can collectively form a mini-batch across clients, but IID forms a mini-batch per client in a decentralized manner.

(ii) IID vs. Random. When data is randomly divided, it will tend to have a distribution close to IID so that there is a minimal performance drop. The averaged gaps for mR@20 are −0.32% and −0.12%, for 'Shard-IID' and 'Dir($\alpha = 10$)', respectively. The results confirm that the random partitioning naively conducted in prior studies is unsuitable for imposing semantic heterogeneity, showing similar results as the IID case.

[2] We attach the detailed description in Appendix A.

[3] The visualization of the clustering is in Appendix A.2.

[4] In Appendix C we attach additional experiments including Convergence behavior, Communication cost, Cluster Imbalance effect, Extension to FL algorithms, and various FL scenarios.

Table 1
Comparison of the performances of PSG methods on the proposed FL benchmark.

R/mR @K	Method	CL[a]	Random	Shard		Dirichlet distribution		
				IID	non-IID	$\alpha = 10(\approx$ IID)	$\alpha = 1$	$\alpha = 0.2$
R/mR @20	IMP	16.54/6.55	12.45/3.08	12.62/3.20	11.26/2.28	12.31/3.36	12.10/2.92	9.31/1.78
	MOTIFS	16.97/7.56	13.54/4.60	13.26/4.64	13.33/4.06	13.33/4.39	13.34/4.09	13.25/4.28
	VCTree	16.80/7.20	12.73/4.38	13.00/4.57	12.49/3.99	13.00/4.42	12.86/4.36	13.06/4.17
	GPS-Net	**18.00/7.83**	**13.93/5.98**	**14.83/6.90**	**14.57/5.90**	**14.88 /6.33**	**14.82/6.16**	**14.38 /5.91**
R/mR @50	IMP	17.87/6.96	13.89/3.44	13.97/3.53	12.57/2.59	13.79/3.73	13.40/3.23	10.83/2.03
	MOTIFS	18.59/8.01	15.07/5.05	14.82/5.06	14.92/4.48	14.77/4.71	14.63/4.44	14.77/4.64
	VCTree	18.54/7.70	14.20/4.75	14.50/4.94	14.04/4.41	14.32/4.82	14.34/4.78	14.51/4.56
	GPS-Net	**19.69 /8.30**	**15.63 /6.51**	**16.42/7.37**	**16.37/6.36**	**16.46/6.74**	**16.34/6.62**	**16.01/6.36**
R/mR @100	IMP	18.37/7.11	14.46/3.56	14.45/3.65	13.06/2.68	14.48/3.89	13.92/3.35	11.25/2.10
	MOTIFS	19.15/8.14	15.64/5.16	15.38/5.20	15.43/4.65	15.33/4.86	15.15/4.62	15.18/4.71
	VCTree	19.02/7.82	14.69/4.87	14.97/5.05	14.62/4.54	14.87/4.97	14.90/4.90	15.03/4.68
	GPS-Net	**20.28/8.47**	**16.34/6.66**	**17.08/7.55**	**16.91/6.49**	**17.10/6.91**	**16.84/6.77**	**16.55/6.51**

Bold refers the best performance and underline denotes the 2nd performance.

[a] For centralized learning (CL) is with a centralized dataset without considering the FL settings.

(iii) IID vs. non-ID. We confirm large performance degradations in most cases. First, in the case of a shard-based partition, the averaged gap for mR@20 is −0.77%. Second, in the case of the Dirichlet distribution-based partition, i.e., comparing Dir($\alpha = 10$) and Dir($\alpha = 0.2$), the averaged gap for mR@20 is shown to be −0.64%. The performance drops from IID to non-IID reveal that PSG methods struggle to aggregate a global model under strong semantic heterogeneity. MOTIFS shows the outliers in mR, where the moderate non-IID case ($\alpha = 1$) compared to the non-IID case ($\alpha = 0.2$) shows minimal differences: 4.09% vs. 4.28% in mR@20, and 4.62% vs. 4.71% in mR@100. It looks unexpected, but it is not a considerable amount. Also, we want to point out that when $\alpha = 10$, which is the IID case, the performance becomes maximized: 4.39% in mR@20 and 4.86% in mR@100, which coincides with our expectations. We conjecture that the behavior at the moderate non-IID can be a little shaky in a few cases, but it finally behaves as expected in the IID case. Although the results may seem unexpected, the differences are not significant. Notably, when $\alpha = 10$, corresponding to the IID case, shows the best performance: 4.39% in mR@20 and 4.86% in mR@100, aligning with our expectations. Based on this observation, we conjecture that the behavior at the moderate non-IID can be a little shaky in a few cases, but it behaves as expected in the IID case.

PSG Model comparisons: We discuss the robustness of the existing PSG methods against semantic heterogeneity. We conclude that IMP is relatively vulnerable in handling semantic heterogeneity in FL, i.e., a large gap of −1.58% for mR@20 is observed when comparing Dir($\alpha = 10$) and Dir($\alpha = 0.2$). It has a smaller model architecture and suffers from the long-tailed problem in the PSG dataset. We conjecture that the aspects of IMP lead to notable performance drops in our non-IID testing. VCTree includes a tree construction process trained via reinforcement learning, resulting in a more complex model structure than MOTIFS. Consequently, in the FL scenario with small-scale client data, its performance is degraded. Because GPS-Net employs key elements, e.g., DMP, NPS-loss, and ARM, to resolve the long-tailed problem, we conjecture that it yields the outperforming results of GPS-Net in our FL benchmarks.

4.3. Extension to FL algorithms

Next, we verify whether the improvements in FL algorithms stay valid in our benchmark. We conducted additional experiments on two FL algorithms employing momentum. Momentum-based update strategies prove effective in maintaining local training closer to the global update direction. By mitigating the adverse effects of data distribution discrepancies, these algorithms can enhance both convergence stability and model performance.

Table 2
Comparison of the FedAvg, FedAvgM, and FedAdam performances of PSG methods.

R/mR@K	Method	FedAvg Shard non-IID	FedAvgM [6] Shard non-IID	FedAdam [7] Shard non-IID
R/mR@20	IMP	11.26/2.28	13.23/3.83 (+1.55%)	13.32/4.78 (+2.50%)
	MOTIFS	13.33/4.06	15.47/5.80 (+1.74%)	**15.89**/5.56 (+1.50%)
	VCTree	12.49/3.99	15.39/5.66 (+1.67%)	15.53/5.09 (+1.10%)
	GPS-Net	**14.57/5.90**	**16.18/5.91** (+0.01%)	15.66/**5.98** (+0.08%)
R/mR@50	IMP	12.57/2.59	14.73/4.24 (+1.65%)	15.03/5.41 (+2.82%)
	MOTIFS	14.92/4.48	17.23/6.23 (+1.75%)	**17.66**/6.01 (+1.53%)
	VCTree	14.04/4.41	17.02/6.10 (+1.69%)	16.95/5.40 (+0.99%)
	GPS-Net	**16.37/6.36**	**18.00/6.33** (−0.03%)	17.31/**6.42** (+0.06%)
R/mR@100	IMP	13.06/2.68	15.32/4.38 (+1.70%)	15.63/5.57 (+2.89%)
	MOTIFS	15.43/4.65	17.83/6.38 (+1.73%)	**18.21**/6.14 (+1.49%)
	VCTree	14.62/4.54	17.58/6.30 (+1.76%)	17.42/5.53 (+0.99%)
	GPS-Net	**16.91/6.49**	**18.68/6.52** (−0.03%)	17.90/**6.55** (+0.06%)

(·) indicates the difference in mR@K when each algorithm is applied compared to FedAvg.

4.3.1. FedAvgM

We present the result of applying FedAvgM [6] in Table 2. FedAvgM utilizes the momentum in updating a global model on the server side and relieves the varying directions of local updates due to the stochastic variance across clients. FedAvgM updates the global model as follows:

$$w_g^{r+1} = w_g^r - v^r, \tag{3}$$

$$v^r = \beta v^{r-1} + \sum_{k=1}^{K} \frac{n_k}{n} \Delta w_k^r \tag{4}$$

where β is the momentum hyperparameter for FedAvgM, n_k is the number of examples, Δw_k^r is the weight update from k's client, and $n = \sum_{k=1}^{K} n_k$.

Results: FedAvgM sufficiently improves the performance of all methods. For R/mR@20, R/mR@50, and R/mR@100, there are average performance improvements of +1.24%, +1.27%, and +1.30% in the Shard-nonIID case, respectively. These performance improvements under FedAvgM can be attributed to FedAvgM's momentum-based updates which help stabilize training by reducing local update oscillations in the optimization process, leading to faster convergence and better generalization across heterogeneous client data. IMP, MOTIFS, and VCTree showed noticeable performance increases, while GPS-Net did not. GPS-Net in the Shard non-IID case shows a negligible gap (i.e., $\leq 0.03\%$), indicating that GPS-Net already incorporates factors that mitigate the effects of heterogeneity.

4.3.2. FedAdam

FedOpt [7] provides a framework for improving optimization, which supports server-side optimization algorithms (e.g., Adam) to enhance

Table 3
Comparisons of FL algorithms on CelebA emotion classification (Smiling vs. Not smiling) task.

Method	Shard		Dirichlet distribution		
	IID	non-IID	$\alpha = 10$	$\alpha = 1$	$\alpha = 0.2$
FedAvg	90.94 (32)	90.43 (52)	91.56 (32)	91.65 (37)	91.03 (47)
FedAvgM	**92.93** (27)	**91.22** (32)	**93.4** (27)	**92.93** (27)	**91.32** (37)
FedAdam	91.84 (52)	91.03 (52)	91.71 (52)	91.77 (52)	90.79 (52)

(·) is the communication rounds to reach 85% Acc.

convergence and stability. This approach deals with divergent client data distributions and fluctuations in client participation rates. FedOpt uses different optimizers in local and global updates. In our case, we utilize the Adam [31] optimizer for global updates:

$$m^{r+1} = \beta_1 m^r + (1-\beta_1)\Delta w^r, \tag{5}$$

$$v^{r+1} = \beta_2 v^r + (1-\beta_2)(\Delta w^r)^2, \tag{6}$$

$$w_g^{r+1} = w_g^r - \eta \frac{m^{r+1}}{\sqrt{v^{r+1}} + \epsilon}, \tag{7}$$

where β_1 and β_2 are the momentum hyperparameters, ϵ is a small constant added to the denominator to ensure numerical stability and prevent division by 0. We present the result of applying FedAdam in Table 2.

Results: FedAdam demonstrated a marginally superior performance improvement compared to FedAvgM. For R/mR@20, R/mR@50, and R/mR@100, there are average performance improvements of +1.30%, +1.35%, and +1.36% in the Shard-nonIID case, respectively. Interestingly, FedAdam shows a noticeable performance improvement when combined with IMP, as shown in the table. IMP has a lower initial performance (based on R/mR@20, R/mR@50, and R/mR@100) when compared to other methods (MOTIFS, VCTree, GPS-Net). However, when combined with FedAdam, it showed the greatest performance improvement (+2.89% in R/mR@100 case). IMP simply learns by iteratively updating relationships between objects. IMP is prone to learning by being overly head-class-biased in class imbalances, and performance degradation is inevitable in tail classes. In this environment, FedAdam has most likely improved its model effectively in the tail class, where losses are concentrated due to the long-tailed problem. Contrary to IMP, GPS-Net has various strategies to solve the long-tailed problem. Similar to the FedAvgM experimental result, GPS-Net showed no significant change in performance.

We can conclude that the enhancement of FL algorithms is effective when dealing with scenarios involving diverse semantic information across clients. Furthermore, the experimental outcomes with IMP and GPS-Net reveal an intriguing connection: the long-tailed problem encountered in scene graph generation tasks shares notable similarities with the data heterogeneity issues faced in FL. In scenarios where scene graph generation tasks must be addressed in a distributed data environment, selecting a scene graph generation method with a strong strategy for handling long-tailed problems or choosing an FL algorithm that effectively deals with data heterogeneity would significantly increase the likelihood of simultaneously tackling both challenges.

5. Additional experiments on CelebA

To demonstrate the generalizability, we also applied our clustering method to the CelebA dataset, which comprises 40 attributes (e.g., Eyeglasses, Wearing hat, Wavy hair), not a specific label.[5]

[5] We attach the details on CelebA experiments in Appendix E.

5.1. Construction of non-IID CelebA dataset

In prior works, the CelebA dataset is distributed only according to identity in a non-IID case, e.g., each client has pictures of the same person, which limits the method for representing a kind of non-IID setting. It also cannot represent practical cases, such as the contents of CCTV or broadcast systems. Therefore, we build a non-IID dataset with the following steps. First, we perform K-Means clustering with 5 clusters. Since the attributes consist of binary values (−1, 1), we do not apply super-classes to preserve the meaning of each attribute. Second, we cluster images and allocate the cluster index to each image. Based on the cluster index of each image, we build non-IID cases by applying data partitioning methods such as shard-based and Dirichlet distribution-based partitions. We evaluate the classification task to check whether each person is smiling in Table 3.

5.2. Analysis

Similar to the result of the PSG task, it shows degradation of performance and a slow convergence rate in the shard non-IID case. For non-IID cases, including shard non-IID and Dirichlet $\alpha = 0.2$, it shows a slower convergence rate than IID cases. For FedAvg, a non-IID environment, including shard non-IID and Dirichlet $\alpha = 0.2$, leads to slower convergence than the IID cases, evidenced by slow convergence of 32→52 under Shard and 32→47 under Dirichlet-based partitions. For FedAvgM, it can also solve non-IID problems, where it reduces the communication cost by 20 more than FedAvg, especially in the Shard non-IID case 52→32. For FedAdam, it shows the slowest convergence rate, conjecturing that it is because of the global learning rate of $1e^{-4}$.

6. Conclusion

Our work takes a decisive step toward closing the gap between federated learning (FL) research and high-level visual understanding by introducing the *first* benchmark framework for multi-semantic vision tasks under controlled data heterogeneity. Existing approaches for generating data heterogeneity rely on single-label datasets and cannot be extended to multiple semantic collections. To overcome this limitation, we propose a clustering-based scheme that groups samples based on the semantics of the images. This design naturally accommodates datasets with multiple labels per image, such as PSG and CelebA. We validate our framework on the PSG dataset and CelebA by partitioning the data according to cluster assignments and measuring convergence behavior. As the data heterogeneity increases, i.e., from non-IID to IID, training converges more slowly and shows lower final accuracy for both datasets, confirming the ability to construct heterogeneity. Furthermore, our benchmark demonstrated consistent trends with prior FL studies, even when extended to various FL scenarios, i.e, changes in participation rates and number of clusters.

While the proposed approach shows significant results, it has limitations. First, determining the optimal number of clusters remains challenging. Second, the method assumes the availability of semantic annotations, which may not always be accessible.

Nevertheless, the proposed benchmark can be adopted in practical use. It can be extended to multi-label and structured data tasks such as multimodal learning, visual question answering, and relation extraction. The framework is highly applicable to domains such as broadcasting and media, where centralized learning is often impractical due to stringent requirements regarding raw data confidentiality, intellectual property, and content ownership. FL with semantic-aware partitioning offers a collaborative model training without compromising sensitive media assets.

CRediT authorship contribution statement

SeungBum Ha: Validation, Formal analysis, Methodology. **Taehwan Lee:** Validation, Methodology, Software, Data curation. **Jiyoun Lim:** Writing – review & editing, Resources, Supervision, Funding acquisition. **Sung Whan Yoon:** Writing – review & editing, Supervision, Funding acquisition, Writing – original draft, Project administration, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. 2021-0-00852, Development of Intelligent Media Attributes Extraction and Sharing Technology) and National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (No. RS-2024-00459023).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.patrec.2025.07.020.

Data availability

The authors do not have permission to share data.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.
[2] A. Krizhevsky, V. Nair, G. Hinton, CIFAR-10 (Canadian institute for advanced research), 2019.
[3] S. Caldas, S.M.K. Duddu, P. Wu, T. Li, J. Konečnỳ, H.B. McMahan, V. Smith, A. Talwalkar, Leaf: A benchmark for federated settings, 2018, arXiv preprint arXiv:1812.01097.
[4] X. Li, K. Huang, W. Yang, S. Wang, Z. Zhang, On the convergence of fedavg on non-iid data, 2019, arXiv preprint arXiv:1907.02189.
[5] D.A.E. Acar, Y. Zhao, R.M. Navarro, M. Mattina, P.N. Whatmough, V. Saligrama, Federated learning based on dynamic regularization, 2021, arXiv preprint arXiv:2111.04263.
[6] T.M.H. Hsu, H. Qi, M. Brown, Measuring the effects of non-identical data distribution for federated visual classification, 2019, arXiv preprint arXiv:1909.06335.
[7] S. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečnỳ, S. Kumar, H.B. McMahan, Adaptive federated optimization, 2020, arXiv preprint arXiv:2003.00295.
[8] Y. Shi, Y. Zhang, P. Zhang, Y. Xiao, L. Niu, Federated learning with L1 regularization, Pattern Recognit. Lett. 172 (2023) 15–21.
[9] T. Lee, S.W. Yoon, Rethinking the flat minima searching in federated learning, in: Forty-First International Conference on Machine Learning, 2024.
[10] P. Foret, A. Kleiner, H. Mobahi, B. Neyshabur, Sharpness-aware minimization for efficiently improving generalization, Int. Conf. Learn. Represent. (ICLR) (2020).
[11] H. Zhu, J. Xu, S. Liu, Y. Jin, Federated learning on non-IID data: A survey, Neurocomputing 465 (2021) 371–390.
[12] I. Achituve, A. Shamsian, A. Navon, G. Chechik, E. Fetaya, Personalized federated learning with gaussian processes, Adv. Neural Inf. Process. Syst. 34 (2021) 8392–8406.
[13] J.H. Lim, S. Ha, S.W. Yoon, MetaVers: Meta-learned versatile representations for personalized federated learning, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 2587–2596.
[14] A.A. Liu, Y. Wang, N. Xu, S. Liu, X. Li, Scene-graph-guided message passing network for dense captioning, Pattern Recognit. Lett. 145 (2021) 187–193.
[15] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.J. Li, D.A. Shamma, et al., Visual genome: Connecting language and vision using crowdsourced dense image annotations, Int. J. Comput. Vis. 123 (2017) 32–73.
[16] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft coco: Common objects in context, in: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, Springer, 2014, pp. 740–755.
[17] J. Yang, Y.Z. Ang, Z. Guo, K. Zhou, W. Zhang, Z. Liu, Panoptic scene graph generation, in: European Conference on Computer Vision, Springer, 2022, pp. 178–196.
[18] M. Zhao, J. Zhang, Panoptic segmentation-based semantic embedding matching model for scene graph generation, Pattern Recognit. Lett. 193 (2025) 56–63.
[19] A. Kirillov, K. He, R. Girshick, C. Rother, P. Dollár, Panoptic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9404–9413.
[20] A. Desai, T.Y. Wu, S. Tripathi, N. Vasconcelos, Learning of visual relations: The devil is in the tails, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 15404–15413.
[21] F. Lv, P. Qian, Y. Lu, H. Wang, Personalized federated learning on long-tailed data via knowledge distillation and generated features, Pattern Recognit. Lett. 186 (2024) 178–183.
[22] X. Chang, P. Ren, P. Xu, Z. Li, X. Chen, A. Hauptmann, Scene graphs: A survey of generations and applications, 2, 2021, arXiv preprint arXiv:2104.01111.
[23] P. Li, Z. Yu, Y. Zhan, Deep relational self-attention networks for scene graph generation, Pattern Recognit. Lett. 153 (2022) 200–206.
[24] X. Li, T. Wu, G. Zheng, Y. Yu, X. Li, Uncertainty-aware scene graph generation, Pattern Recognit. Lett. 167 (2023) 30–37.
[25] D.J. Kim, T.W. Ke, X.Y. Stella, Local pseudo-attributes for long-tailed recognition, Pattern Recognit. Lett. 172 (2023) 51–57.
[26] D. Xu, Y. Zhu, C.B. Choy, L. Fei-Fei, Scene graph generation by iterative message passing, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5410–5419.
[27] R. Zellers, M. Yatskar, S. Thomson, Y. Choi, Neural motifs: Scene graph parsing with global context, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5831–5840.
[28] K. Tang, H. Zhang, B. Wu, W. Luo, W. Liu, Learning to compose dynamic tree structures for visual contexts, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 6619–6628.
[29] X. Lin, C. Ding, J. Zeng, D. Tao, Gps-net: Graph property sensing network for scene graph generation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3746–3753.
[30] R. Girshick, Fast r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1440–1448.
[31] D.P. Kingma, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.