International Journal of Network Dynamics and Intelligence Scilight

Article

Federated Bimodal Graph Neural Networks for Text-Image Retrieval

Xueming Yan [1,2], Chuyue Wang [1], and Yaochu Jin [3,*]

[1] School of Information Science and Technology, Guangdong University of Foreign Studies, Guangzhou, 510006, China
[2] Guangdong Engineering Research Center of Data Security Governance and Privacy Computing, Guangzhou, 510006, China
[3] School of Engineering, Westlake University, Hangzhou, 310030, China
[*] Correspondence: jinyaochu@westlake.edu.cn

Received: 24 December 2024
Accepted: 20 March 2025
Published: 27 June 2025

Abstract: Text-image retrieval is a key challenge in computer vision and natural language processing, aiming to retrieve the most semantically relevant image or text given a query in the opposite modality. However, growing privacy and security concerns make traditional centralized learning approaches increasingly unsuitable for handling sensitive multimodal data. In this paper, we propose FedBi-GNNs, a federated learning framework for bimodal graph neural networks, which enables collaborative training across decentralized clients without sharing private data. Each client independently constructs heterogeneous graphs from local text and image data and learns correspondences via bimodal graph matching. These local representations are then aggregated at a central server using a heterogeneous federated aggregation scheme. Empirical results on the MSCOCO benchmark demonstrate that FedBi-GNNs significantly outperform existing state-of-the-art methods, offering improved retrieval accuracy, enhanced privacy preservation, and greater robustness to data heterogeneity across clients.

Keywords: federated learning; bimodal graph neural networks; text-image retrieval; graph matching

1. Introduction

In the ever-evolving landscape of computer vision and natural language processing, text-image retrieval remains a pivotal task with wide-ranging applications. The goal is to retrieve the most relevant textual or visual content from a dataset given a query in the form of text or an image [1, 2]. One of the primary challenges lies in bridging the semantic gap between visual and textual modalities to enable accurate similarity comparison between text-image pairs. To tackle this issue, earlier approaches [3−5] typically projected the global feature vectors of images and texts into a shared latent space, where semantically similar pairs are represented by closer embeddings.

However, when the input data involves complex scenes (e.g., with multiple entities), these methods often yield suboptimal retrieval performance. As a result, growing research attention has been directed toward capturing local correspondences within bimodal data by extracting fine-grained features [6]. For example, SCAN [7] employs a stacked cross-attention mechanism that assigns varying weights to different image regions for representing individual text words, and vice versa. Building upon this, PFAN [8] introduces positional embeddings to guide the learning of semantic alignments. In parallel, several methods explore intra-modality and inter-modality fusion techniques to bridge the feature disparity between visual and textual modalities. For instance, MMCA [9] designs a network that achieves both intra- and inter-modality fusion through attention mechanisms. SHAN [10] further proposes a progressive hierarchical alignment fusion network to estimate the similarity between text-image pairs more effectively.

Recent studies have also made significant progress in incorporating graph neural networks (GNNs) into text-image retrieval. GNNs have emerged as powerful tools for modeling both global and local structures in bimodal data represented as graphs [11]. For example, SGM [12] and LGSGM [13] focus on extracting global and local features from pre-generated scene graphs of images and texts. Similarly, GSMN [14] and SGRAF [15] construct graphs using

the stacked cross-attention mechanism introduced by SCAN [7]. These approaches aim to generate both local and global similarity vectors and infer final similarity scores through graph-based reasoning. By leveraging graph structures, they enhance the model's ability to capture fine-grained and holistic information, leading to substantial improvements in retrieval performance. However, most existing methods rely on centralized datasets, where all image and text data are stored on a single central server, as illustrated in Figure 1a. This centralized paradigm introduces several limitations, including data privacy risks, security concerns, and practical difficulties in accessing and integrating data distributed across multiple locations [16].

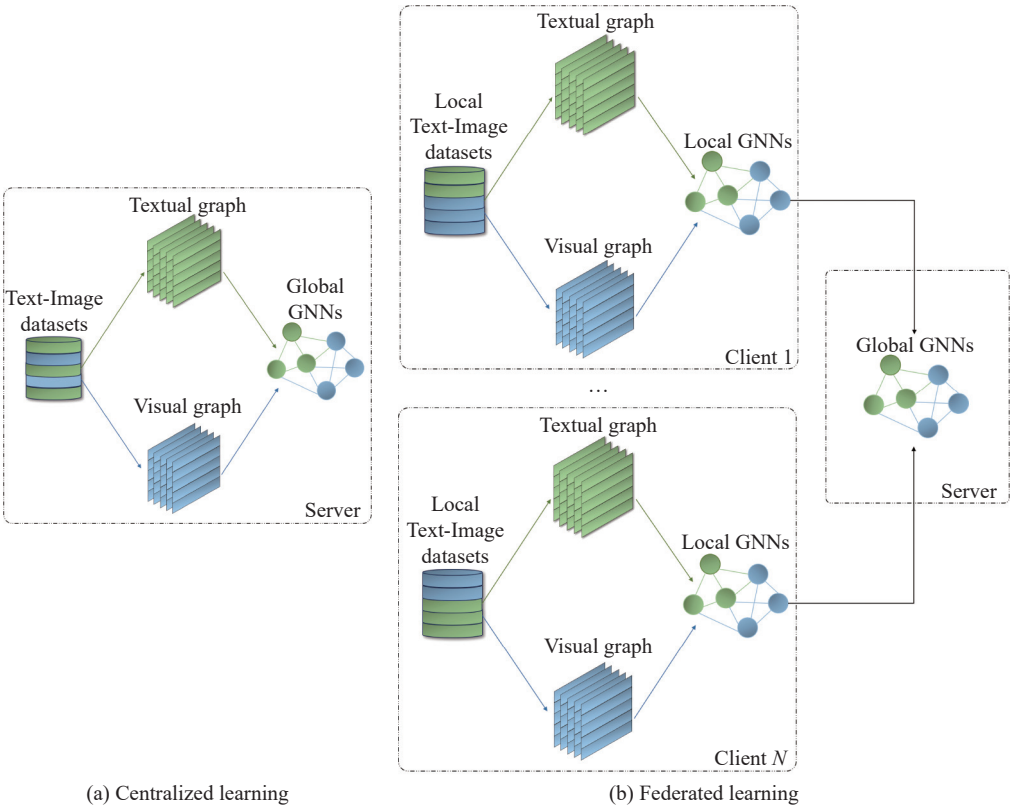

Figure 1. Comparisons between centralized GNNs and federated learning of bimodal GNNs model for text-image retrieval.

To address these issues, we introduce federated learning into the training of bimodal graph neural networks (FedBi-GNNs) for text-image retrieval, as illustrated in Figure 1b. In our FedBi-GNNs, bimodal GNNs are employed to process text-image data, where each image and each text sample is represented as an individual graph. Moreover, federated models for text-image retrieval must accommodate the increased complexity of bimodal data, which poses greater challenges than unimodal models in terms of both model capacity and computational overhead [11, 17–20]. Our FedBi-GNNs not only preserves data privacy but also improves retrieval accuracy and efficiency by leveraging the data diversity available across decentralized clients. To the best of our knowledge, this is the first work to integrate federated learning with bimodal GNNs for text-image retrieval, offering an efficient and privacy-preserving solution for training retrieval models under distributed data storage.

The main contributions of this work are as follows:

- We propose the FedBi-GNNs, a federated bimodal graph neural network framework for text-image retrieval, which enables federated learning by aggregating model parameters on a central server without accessing clients' bimodal data.
- We design a bimodal graph matching mechanism within cross-modal GNNs, allowing the model to learn meaningful correspondences between textual and visual representations through heterogeneous graph alignment.
- Extensive experiments demonstrate that FedBi-GNNs significantly improves training efficiency and retrieval performance while preserving data privacy, outperforming previous state-of-the-art text-image retrieval methods.

2. Related Work

Text-image retrieval aims to retrieve the most relevant text description given an image or retrieve the most relevant image given a text. A grand challenge in this field is how to extract semantic features from both images and texts

effectively. After obtaining image and text features, feature alignment is also a critical step. Existing methods mainly fall into two categories: global feature alignment and local feature alignment. Global feature alignment methods typically employ a two-branch structure to learn overall feature representations of images and texts separately. Then, they map these representations to a common space to compute similarity. The issue with such methods is that they heavily rely on global information while overlooking fine-grained correspondences between local image regions and textual keywords. For instance, VSE++ [21] employs global feature alignment by learning holistic representations of images and texts through two separate networks, followed by similarity computation. However, such global matching cannot effectively model the fine-grained relationships between specific regions in images and keywords in texts, resulting in less accurate matching results.

To enable finer-grained matching, recent studies have started to consider local feature alignment. A common approach is to introduce attention mechanisms [7] to learn attention weights between different image regions and words in the text, achieving local matching. However, simple attention mechanisms may not adequately model the optimal matching between regions and words. Therefore, some studies propose more sophisticated mechanisms to identify the best local alignment. For example, VSRN [22] introduces graph structures to model relationships between objects and words. It employs graph convolutional networks to achieve relation-aware feature aggregation, resulting in better local feature alignment. SHAN [10] adopts a stage-wise matching approach, progressively matching local-to-local, global-to-local, and global-to-global to gradually acquire semantic associations between images and texts, enabling more accurate fine-grained matching. GSMN [14] considers matching not only between individual regions and words but also multi-granularity local alignment between textual segments at different levels (words, phrases, etc.) and image regions. Based on the idea of GSMN, we adopt the hierarchical matching idea to perform bidirectional retrieval between images and texts represented as graphs.

Furthermore, there have been some recent efforts to employ federated learning in text-image retrieval. For example, FedCRM [23] introduces federated learning for text-image retrieval by performing local feature extraction and model updates on local devices, thereby achieving cross-modal retrieval capabilities while safeguarding user privacy. However, when compared to mainstream text-image retrieval methods, the performance of FedCMR is not outstanding. This can be attributed to the fact that the local model DSCMR used by FedCRM is based on earlier methods that map global feature vectors of images and texts into a common latent space for comparing semantic similarity. Such methods struggle to capture the local similarities between the two modalities effectively. In addition to FedCRM, other federated learning methods, such as FedAVG (Federated Averaging) [24] and FedProx (Federated Proximal) [25], have also been applied to text-image retrieval. These methods enable cross-device model training and optimization through local model updates and parameter aggregation on local devices. The integration of federated learning allows text-image retrieval tasks to leverage decentralized data for model training while maintaining privacy, thereby enhancing retrieval accuracy and performance.

3. The Proposed Method

In this section, we first present the overall framework of our FedBi-GNN for text-image retrieval. The FedBi-GNN trains bimodal GNNs by leveraging highly decentralized text-image data across clients to learn semantic correspondences between textual and visual modalities. We then elaborate on the construction of heterogeneous graph representations and the bimodal graph matching process performed locally within each client. Finally, we introduce the federated heterogeneous aggregation strategy, which is designed to evaluate whether the global model can effectively integrate client updates to enhance local bimodal GNNs and support continued federated training.

3.1. Our Framework

Figure 2 demonstrates the framework of FedBi-GNN, which consists of a central server and a large number of client users. Each client contains two graphs, including a textual graph composed of sentence words and a visual graph composed of image regions. In particular, each client learns the semantic correspondence between texts and images and the GNN model from its local graphs, and uploads the gradient information to the central server. The central server is responsible for aggregating the gradient information received from multiple clients and sending the aggregated gradient information back to them to coordinate their work in the model learning process.

The pseudocode of the proposed FedBi-GNN is shown in Algorithm 1, including four main components: heterogeneous graphs representation (line 4), bimodal graph Matching (lines 5-6), federated heterogeneous aggregation (lines 10-11), and update for local bimodal GNNs (lines 12-14). Assuming there are N clients, denoted as $C = C_1, C_2, \cdots, C_N$, and where C_i represents the i-th client. Each client has a data set $D = D_1, D_2, \cdots, D_N$, where $D_i^k = \{Text, Image\}_i^k$ represents the k-th instance in client C_i, consisting of textual and image modalities. Initially, the text is encoded into word-level feature vectors t_j through GloveRNNEncoder, and the image is represented as

feature vectors v_i after being partitioned into regions by the object detection algorithm. Then, the two vectors are dot multiplied to obtain the word-region matching matrix A, which is then matrix-multiplied with the two feature vectors respectively to obtain the weighted combination sum of them. The "①" in Figure 2 represents the operation of calculating the similarity between computation blocks (corresponding to Formulas 12 and 13), and the "②" represents the operation of vector dimensionality expansion. After operations "①" and "②", we can obtain the graph structure representations of text and image, namely text graph and image graph. Next, we apply bimodal graph neural networks to the textual graph and the visual graph to model the semantic correspondence between nodes in the graphs, where we use two GCNs to predict T2I (text to image) and I2T (image to text) similarities on the textual graph and visual graph respectively, and take the sum of the outputs of the two GCNs as the overall similarity score. Finally, according to the obtained similarity scores, we compute the overall loss function of the local bimodal GNNs. We use the loss function to compute gradients for the local model, which will be further uploaded to the server for aggregation. In the proposed FedBi-GNN, the server aims to coordinate all clients and compute global gradients to update model parameters on these clients. In each round, the server wakes up a certain number of user clients to compute gradients locally and send them to the server. After receiving these users' gradients, the aggregator on the server aggregates these local gradients into a unified gradient. Then the server sends the aggregated gradient back to each client for local parameter update. This process will be executed iteratively until the model converges.

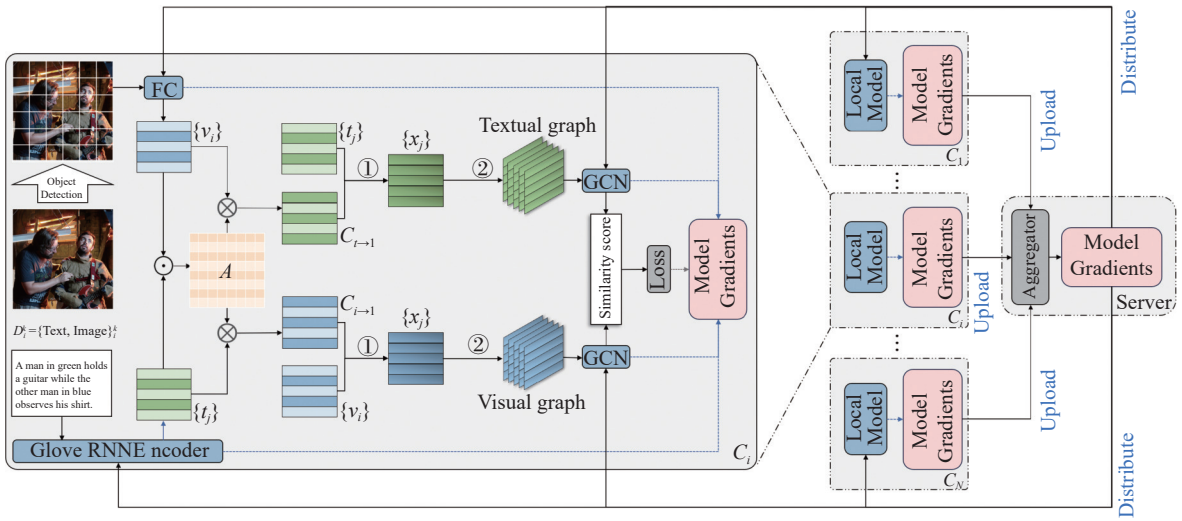

Figure 2. The framework of the proposed FedBi-GNNs.

Algorithm 1: The pseudocode of the proposed FedBi-GNNs

Require The set of clients, $C = \{C^1, C^2, \cdots, C^N\}$; The set of dataset, $D = \{D^1, D^2, \cdots, D^N\}$; The server model, $ M_S $; The number of communication round, $ T $; The number of selected clients, $ K $;

For each round $t = 1, \cdots, T$

1: $C_t, D_t \leftarrow$ random set of K clients

For each client $c \in C_t$ **in parallel**

2: Constructing image graph and text graph by Equation (1)-Equation (7)

3: Performing graphs matching in both I2T and T2I by Equation (8)-Equation (15)

4: Validating the model and obtaining the r_i score

5: Obtaining the trained local model parameters θ_i

6: Send r_i and θ_i to M_S

7: **end for**

8: Obtains weights w_i by Equation (17)

9: Obtains aggregated global parameters θ_{global} by Equation (16)

10: Update θ_{global} to M_S

11: Obtains the updated local model parameters θ_i' by Equation (18)

12: Update θ_i' to M_{C^i}

13: **end for**

3.2. Heterogeneous Graphs Representation

For heterogeneous image and text data, we adopt similar approaches to construct graphical representations. Images are partitioned into regions as nodes in image graphs, while texts are partitioned into words as nodes in text graphs. Next, we will present how images and texts are represented as image and text graphs, respectively.

3.2.1. Image Graph

We represent each image as a fully connected, undirected graph when constructing the visual graph. The graph consists of 36 nodes, each corresponding to a salient region detected by Faster R-CNN [26], with every node connected to all others. The extracted image features are then passed through a fully connected layer to obtain $ D $-dimensional image feature vectors, as follows:

$$v_i = Wo_i + b \tag{1}$$

Here, v_i represents the feature vector of the i-th region, o_i denotes the region information and feature vector of the i-th salient entity, W is the weight matrix of the fully connected layer, and b is the bias vector. By multiplying the region information and feature vector of each salient entity with the weight matrix and adding the bias vector, we obtain the feature vector for each salient entity, which serves as the feature representation of the image. Based on previous experience [7], we set the number of extracted salient regions k for each image to 36, and the dimension $ D $ of the image feature vectors to 1024. Besides, we employ the Faster R-CNN object detection model to detect salient entities in image regions. Ultimately, Faster R-CNN identifies k salient entities, denoted as $O = o_1, o_2, \cdots, o_k \in \mathbb{R}^{k \times 2048}$, within each image and extracts their corresponding regions.

Additionally, we employ polar coordinates to model the spatial relationships within each image. Specifically, the polar coordinates are computed based on the centers of the bounding boxes for each pair of regions, as follows:

$$\begin{aligned} \rho &= \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}, \\ \theta &= \arctan 2(y_2 - y_1, x_2 - x_1) \end{aligned} \tag{2}$$

Here, (x_i, y_i) and (x_j, y_j) represent the center coordinates of a region pair. ρ represents the distance between the region pair, while θ denotes the direction between the region pair, with positive values indicating the counterclockwise direction from the positive x-axis as the reference. By calculating the polar coordinates of region pairs, we transform spatial relationships into a set of values. This transformation helps capture semantic and spatial relationships between different regions, as these relationships and attributes are often associated with the position and orientation of objects. Positional information aids in estimating the strength of relationships, with closer distances generally corresponding to stronger relationships. Directional information assists in determining the type of relationships, such as whether the directional relationship between two regions is "close" or "far". After calculating the polar coordinates (ρ, θ) based on the centers of bounding boxes for region pairs, we use these values to set the edge weight matrix W_e. Specifically, for each region pair (i, j), we compute their polar coordinates (ρ_{ij}, θ_{ij}). These polar coordinates are then used as the elements of the edge weight matrix W_e, i.e., $W_{e_{[i,j]}} = (\rho_{ij}, \theta_{ij})$. In this way, polar coordinates effectively represent the spatial relationships between each region pair and serve as edge weights for constructing the fully connected graph representing the image.

3.2.2. Text Graph

In this section, we adopt an approach similar to that used for constructing image graphs, where each word is treated as a node to form an undirected, fully connected textual graph $G^t = (V^t, E^t)$, analogous to the image graph representation.

To represent a given text U consisting of k words, we first extract word-level textual features involving converting each word into a 1024-dimensional vector using one-hot encoding based on the entire vocabulary of the training dataset, and then we transform the one-hot encoding into fixed and adjustable vectors using the pre-trained word embedding model GloVe [27] and learnable embedding layers. These two vectors are concatenated to obtain the final word representations. Subsequently, we input these vectors into a bidirectional gated recurrent unit (Bi-GRU) to enhance the word-level representations by capturing contextual information in both forward and backward directions of the sentence:

$$\begin{aligned} \overrightarrow{\mathbf{h}}_j &= \overrightarrow{\mathrm{GRU}}\left(\mathbf{w}_j, \overrightarrow{\mathbf{h}}_{j-1}\right) \\ \overleftarrow{\mathbf{h}}_j &= \overleftarrow{\mathrm{GRU}}\left(\mathbf{w}_i, \overleftarrow{\mathbf{h}}_{j-1}\right) \end{aligned} \tag{3}$$

Here, $\overrightarrow{\mathbf{h}}_j$ and $\overleftarrow{\mathbf{h}}_j$ represent the forward and backward GRU hidden states, respectively, with their dimension set to $D = 1024$. The textual feature for the j-th word is obtained by averaging these forward and backward GRU hidden states:

$$w_j = \frac{\overrightarrow{\mathbf{h}}_j + \overleftarrow{\mathbf{h}}_j}{2} \tag{4}$$

Then, we utilize these word vectors to construct a textual undirected fully connected graph. When constructing an undirected fully connected graph for the text, each word is considered a node, and an edge exists between any two nodes i and j, denoted as:

$$E^{t}_{(i,j)} = 1 \tag{5}$$

This approach captures all semantic relationships within the text. Clearly, we can represent this connectivity between word nodes using an adjacency matrix A, which is an identity matrix. Subsequently, we compute the vector similarity s_{ij} between two words:

$$s_{ij} = \frac{\exp\left(\lambda w_i^T w_j\right)}{\sum_{j=1}^{m} \exp\left(\lambda w_i^T w_j\right)} \tag{6}$$

where s_{ij} represents the similarity between node i and node j, which indicates the relationship between words. λ is a scaling factor. We also use a weight matrix W_e to denote edge weights in the graph. Specifically, we obtain W by taking the element-wise product of the similarity matrix S and the adjacency matrix A, followed by L2 normalization:

$$W_e = |S \odot A|_2 \tag{7}$$

Here, S is the similarity matrix consisting of the similarities s between each pair of nodes, and each element in W_e indicates the degree of semantic dependence for the corresponding edge.

3.3. Bimodal graph Matching

We construct heterogeneous image and text graphs to model spatial, semantic and contextual relations within and across modalities. Next, we match the bimodal graphs in two steps: node-level matching and structure-level matching. Graph matching aligns image and text representations by combining node semantic similarities and structural dependencies, laying the foundation for subsequent text-image retrieval.

3.3.1. Node-Level Matching

Node-level matching involves the process of establishing correspondences between individual nodes in the visual graph and the textual graph to learn their relationships. For instance, let's consider node-level matching on the textual graph.

In the initial step, we employ a bidirectional cross-attention mechanism to create one-to-one correspondences between nodes in the textual graph, which are formed by textual words, and nodes in the visual graph, formed by image regions. This process is illustrated in the figure above. Mathematically, we begin by calculating the word-region matching matrix based on the word vectors representing textual nodes and the region vectors representing image nodes. This matrix is denoted as:

$$A = t_i v_j^T \tag{8}$$

where $A \in \mathbb{R}^{k \times n}$ is the word-region matching matrix, representing the similarity between image and text nodes. A_{ij} represents the semantic similarity between the $ i $-th region and the j-th word. The similarity values in A measure the correspondence between image nodes and each textual node. We then aggregate all visual nodes into a weighted combination of their feature vectors, where the weights are calculated based on similarities. This process can be expressed as:

$$C_{t \longrightarrow i} = SoftMax_j(\lambda A)\, v_j \tag{9}$$

where λ is a scaling factor that focuses on the matching nodes.

Next, we divide the i-th feature of the textual node and its corresponding aggregated image node into j blocks, represented as $\left[t_{i1}, t_{i2}, \cdots, t_{ij}\right]$ and $\left[c_{i1}, c_{i2}, \cdots, c_{ij}\right]$, and calculate the inter-block similarity between each pair of blocks. For example, the similarity calculation for the r-th block is:

$$x_{ir} = cos(t_{ir}, c_{ir}) \tag{10}$$

where x_{ir} is a scalar value, and $cos(\cdot)$ represents cosine similarity calculation. By concatenating all block similarities, we obtain the matching vector for the i-th textual node:

$$x_i = x_{i1} \,||\, x_{i2} \,||\, \cdots \,||\, x_{ij} \tag{11}$$

where "||" represents concatenation. In this way, each textual node is associated with its matching visual nodes, which will be propagated to its neighbors in the structural-level matching to guide the learning of phrase correspondences. Similarly, given an image graph, node-level matching will be performed on each node. The corresponding textual nodes will be associated differently:

$$C_{i\longrightarrow t} = SoftMax_i\left(\lambda A^{'}\right) t_i \tag{12}$$

where A' represents the representation of $v_j t_i^T$. Then, each visual node and its associated textual nodes will be processed through multiple block modules to generate matching vectors x.

3.3.2. Structure-Level Matching

Structure-level matching operates on the node-level matching vectors represented as x and propagates these vectors along with the graph edges to neighboring nodes. This approach is advantageous for learning local correspondences, as neighboring nodes can offer valuable contextual relationships. To be more precise, structure-level matching updates the matching vector for each node by incorporating matching vectors from neighboring nodes using Graph Convolutional Networks (GCNs). The GCN layer utilizes K kernels, which are responsible for learning the aggregation of neighboring matching vectors. This process is expressed as follows:

$$\hat{x}_i = \|_{k=1}^{K} \sigma\left(\sum\nolimits_{j\in N_i} W_e W_k x_j + b\right) \tag{13}$$

where N_i represents the neighborhood of the i-th node and W_e represents the edge weights. W_k and b are parameters that are learned for the k-th kernel. K kernels are applied, and the output of the spatial convolution is defined as the concatenation of the outputs from these kernels. This results in convolutional vectors that capture connected node correspondences forming local phrases.

By propagating neighboring node correspondences, phrase correspondences can be inferred. These phrase correspondences are then used to calculate the overall matching score for the text-image pair. The convolved vectors are fed into a multilayer perceptron (MLP) to jointly consider all the learned phrase correspondences and infer the global matching score. This global matching score indicates the degree of matching between one structural graph and another. The process is formulated as follows:

$$\begin{aligned} s_{t\to i} &= \frac{1}{n}\sum_{k} W_s^t\left(\sigma\left(W_h^t \hat{x}_k + b_h^t\right)\right) + b_s^t \\ s_{i\to t} &= \frac{1}{m}\sum_{j} W_s^i\left(\sigma\left(W_h^i \hat{x}_j + b_h^i\right)\right) + b_s^i \end{aligned} \tag{14}$$

where W_s and b_s represent the parameters of the MLP, which consists of two fully connected layers. $\sigma(\cdot)$ represents the hyperbolic tangent (tanh) activation function. Performing structure-level matching on both the visual graph and textual graph allows the model to learn complementary phrase correspondences between the two modalities. The text-image pair's overall matching score, denoted as sim, is computed as the sum of matching scores from both directions in the following.

$$sim = s_{t\to i} + s_{i\to t} \tag{15}$$

3.4. Federated Heterogeneous Aggregation

In the proposed FedBi-GNNs, multiple clients collaborate with a trusted central server to collectively train a machine learning model. During each communication round, selected clients individually compute updates to the model using their local datasets and then upload these updated model parameters to the server for aggregation. When aggregating model parameters in federated learning, the server frequently assigns weights to each client's contribution based on their performance in the current round. These weights are then employed for weighted aggregation. During each round of federated training, the weights for aggregation are usually determined by considering factors such as the amount of data that each client contributed and the quality of their training results. This weighting mechanism ensures that clients with larger datasets or superior performance significantly impact the model's updates during the aggregation process.

Specifically, assume K clients are participating in training, with model parameters $\theta_1, \theta_2, \cdots, \theta_K$ for each client, and corresponding data size $D_1, D_2, \cdots, D_K$ and training result $r_1, r_2, \cdots, r_K$. The global model parameters can be computed using weighted averaging:

$$\theta_{global} = \sum_{i=1}^{K} w_i \theta_i \tag{16}$$

where w_i denotes the weight for the i-th client, which can be calculated based on the client's data size and training result. A common approach is to define the weight as:

$$w_i = \frac{D_i r_i}{\sum_{k=1}^{K} D_k r_k} \tag{17}$$

where D_i is the data size and r_i is the training result of the i-th client. Using weighted aggregation, weights can be computed based on each client's data size and training results to reflect their contributions to the global model better. This can better balance each client's contributions in the parameter aggregation process, so that clients with better performance have more influence on the model update, thereby improving the effectiveness of federated learning.

3.5. Update for Local Bimodal GNNs

Once the global model parameters θ_{global} are aggregated, they need to be sent back to each client to update their local models. This process can be expressed as:

$$\theta_i' = \theta_{global} + \Delta\theta_i \tag{18}$$

where θ_i' represents the updated value of the local model parameters for the i-th client, θ_{global} represents the global model parameters, and $\Delta\theta_i$ represents the difference between the i-th client's local model parameters and the global model parameters.

Specifically, $\Delta\theta_i$ can be obtained by calculating the difference between each client's local model parameters and the global model parameters, typically using the following formula:

$$\Delta\theta_i = \alpha\left(\theta_{global} - \theta_i\right) \tag{19}$$

where α represents the learning rate for parameter updates, controlling the speed of updating the local model parameters.

Using these formulas, the global model parameters θ_{global} are updated by adding the differences $\Delta\theta_i$, resulting in the updated local model parameters θ_i'. Subsequently, θ_i' is sent back to the corresponding client to update its local model parameters, completing one round of federated learning iteration.

4. Experiments

4.1. Datasets and Baselines

We evaluate our model using the widely used MSCOCO dataset [28], which consists of multiple text-image pairs. Each image is associated with five texts describing it. MSCOCO comprises a total of 123,287 images and 123,287 × 5 = 616,435 sentences. Following the conventions established in previous research [7, 10, 12−14], we split the dataset into 113,287 training images, 5,000 validation images, and 5,000 test images. In real-world scenarios, data across clients are often non-independent and identically distributed (non-IID) [29, 30]. To simulate this non-IID property, there are three different ways to create non-IID data across clients [31]: feature distribution skew, label distribution skew, and quantity skew. In our experiments, we use quantity skew to simulate the non-IID property of data across different clients in federated learning. For each client, we allocate a random subset of the full training set as the local training set while retaining the full validation and test sets for local validation and testing purposes. As shown in Table 1, we allocate the entire MSCOCO training set to client 1, randomly sample 80% of the full training set as client 2's training set, and randomly sample 60% of the full training set as client 3's training set.

Table 1 Data allocation of MSCOCO training dataset among clients

	Client 1	Client 2	Client 3
Percentage	100%	80%	60%
Number of Texts	566,435	453,150	39,860
Number of Images	113,287	90,630	7,972

In the experiments, we compare the proposed FedBi-GNN with a series of existing methods to assess its performance in text-image retrieval tasks, mainly including:

- Classic cross-attention methods: some traditional methods that use cross-attention mechanisms to align and retrieve information between text and image modalities, such as SCAN [7], PFAN [8], MMCA [9], and SHAN [10].

● Centrally trained graph matching methods: These methods, such as SGM [12] and GSMN [14], focus on graph-based representations for matching text and images. They often involve training a centralized model to perform the matching task.

● SGRAF [15]: SGRAF is a current state-of-the-art method (SOTA) with the highest retrieval performance in text-image retrieval.

4.2. Experiment Setting

We evaluate the performance of each local model and the global model using Recall@K (K = 1, 5, 10) as the evaluation metric. Recall@K measures the proportion of relevant instances that are retrieved among the top-K results. Specifically, we denote the scores of retrieved instances in the top 1, 5, and 10 results as R@1, R@5, and R@10, respectively. To be consistent with previous studies, we assess our method's effectiveness in both text retrieval (identifying the most relevant text given an image) and image retrieval (finding the matching image for a specific text description) tasks. A higher recall value indicates better performance. Additionally, to show the overall matching performance, we employ the "rSum" metric, which sums up the recall values across all evaluation points, to compare our model with other state-of-the-art methods. Finally, we evaluate the model's performance by taking the average over 5-fold cross-validation on 1000 test images.

We set the number of federated training rounds to 30. In each training round, there are 3 local clients participating, with each client performing 1 epoch of training. We choose to validate the local model on the validation set after each epoch of training on the clients and validate the federated model after each round of model aggregation. Finally, we select the model with the highest rSum on the validation set for testing. We use MSCOCO datasets and conduct 30 rounds of training on 1 NVIDIA GeForce RTX 3070Ti GPU. When extracting feature representations for the image modality, we use Faster-RCNN to extract feature vectors of image regions, obtaining 36 salient regions. The dimensions are converted from 2048 to 1024 through a fully connected layer. For the text modality, we set the word vector dimension to 300 and obtain 1024 word embedding dimensions through Bi-GRU. We use the Adam optimizer for learning optimization, with an initial learning rate set to 0.0005, decayed by 10% every 10 epochs, and a mini-batch size of 64. In structure-level matching, we use a spatial graph convolutional layer with 8 convolutional kernels, each being 32-dimensional. After that, we input each node in the graph into two fully connected layers and use a tanh activation function to infer the matching score. The triplet loss margin in the optimization function is set to 0.2.

4.3. Experiment Result Analysis

The comparison experimental results on MSCOCO are shown in Table 2. Our method ranks among the top 2 in terms of R@1, R@5, and R@10 for both image-to-text (I2T) and text-to-image (T2I) retrieval. Compared to single-step matching methods like SCAN and PFAN, the progressive graph matching approach of FedBi-GNNs can better model semantic associations between texts and images. Specifically, it improves R@1 over SCAN by 7.5% and 5.4% for I2T and T2I retrieval, respectively; and over PFAN by 3.7% and 2.6%, respectively. Additionally, on the important rSum metric, our method improves over SCAN and PFAN by 19.3% and 9.0%, respectively. Clear performance gains can also be observed when compared to multi-step matching methods MMCA and SHAN based on SCAN. Specifically, we achieve improvements of 6.0% and 9.4% on R@1 over MMCA for I2T and T2I retrieval, respectively, and 3.4% and 1.6% over SHAN.

Table 2 Text-image retrieval results on MSCOCO

Methods	Image-to-Text			Text-to-Image			rSum
	R@1	R@5	R@10	R@1	R@5	R@10	
SGM	73.4	93.8	97.8	57.5	87.3	94.3	504.1
GSMN	78.4	96.4	98.6	63.3	90.1	95.7	522.5
SGRAF	79.6	96.2	98.5	63.2	90.7	96.1	524.3
SCAN	72.7	94.8	98.4	58.8	88.4	94.8	507.9
PFAN	76.5	96.3	**99.0**	61.6	89.6	95.2	518.2
MMCA	74.2	92.8	96.4	54.8	81.4	87.8	487.4
SHAN	76.8	96.3	98.7	62.6	89.6	95.8	519.8
FedBi-GNNs(Ours)	**80.2**	**96.6**	98.8	**64.2**	**91.2**	**96.3**	**527.2**

Moreover, unlike centralized graph matching methods like SGM and GSMN, FedBi-GNNs adopts a federated learning framework that exploits data from different clients for collaborative training while protecting data privacy, improving model generalization. Our method outperforms SGM on R@1 by 6.8% and 6.7% for I2T and T2I retrieval, respectively, and GSMN by 1.8% and 0.9%, respectively. It also improves over the best rSum result from SGRAF by 3.1%. Finally, compared to the state-of-the-art SGRAF model, we achieve superior performance across all metrics. Specifically, we obtain improvements of 0.6% and 1% on R@1 for I2T and T2I retrieval, respectively.

We also improve rSum by 3.1% over SGRAF. In general, by adopting a federated learning framework, FedBi-GNNs enables separate local client training and avoids centralized data storage for better privacy and security. It also collaboratively trains on distributed data to improve generalization.

To better demonstrate our experimental results, we also visualized the I2T and T2I retrieval results on MSCOCO, as shown in Figure 3 and Figure 4. These show that our approach consistently achieves a high-ranking retrieval of the ground truth. We can clearly observe semantic relevance between the images and texts even for retrieved results outside of ground truth matches in the dataset.

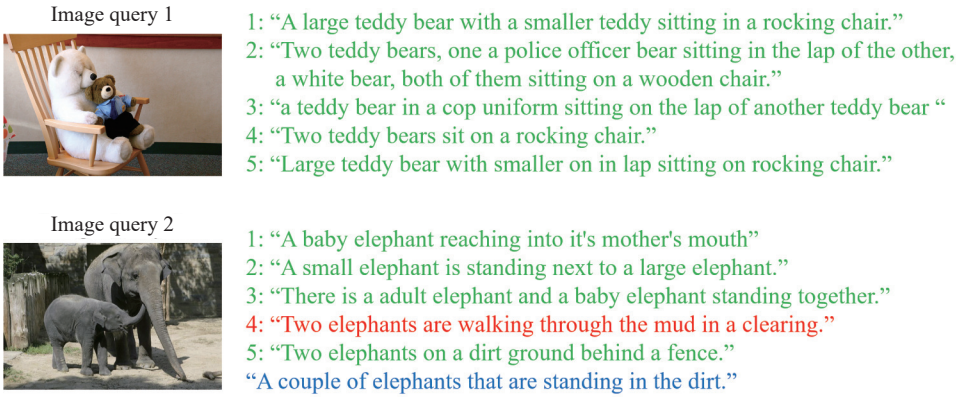

Figure 3. Visualization of text retrieval results on MS-COCO for given image queries. Each image has five ground truth-matching descriptions. For each sentence query, we show the top 5 ranked images from ranks 1 to 5. Real matches are labeled in green, false matches are in red, and real matches outside the top 5 are in blue.

Figure 4. Visualization of image retrieval results on MSCOCO for given sentence queries. Each sentence describes one ground truth image. We show the top 5 ranked images for each query from left to right. Real matches are labeled in green, and false matches in red.

4.4. Effect of Federated Learning

To validate the efficacy of our proposed FedBi-GNNs framework, we trained both federated and single-client models on the MSCOCO dataset for comparison in Table 3. In the federated setting, we employed three clients, each of which was trained locally for one epoch per round using its own local data. The server aggregated model parameters from the clients after each round. A total of 30 training rounds were conducted. In the single-client setting, a model was trained on a single client using the entire MSCOCO training set. This model also trained for one epoch per round without any server communication or parameter aggregation. We recorded the performance of both models at each round using metrics like R@1 and R@5 and compared their best results.

As shown in Table 3, the federated model outperformed the single-client model across all evaluation metrics. Specifically, it achieved a 3.6% improvement in R@1 and an 8.1-point increase in rSum for the image-to-text (I2T) retrieval task. These performance gains can be attributed to the improved generalization enabled by aggregating knowledge from multiple clients. The single-client model, trained on a single dataset, suffered from limited data diversity and thus exhibited weaker generalization. In contrast, the federated model benefits from the integration of sample distributions from different clients, leading to more robust and transferable representations. Moreover, these

improvements are achieved without sharing raw data, highlighting the potential of federated learning for privacy-preserving multimodal applications.

Table 3 Performance comparison between federated model and single client settings

Setting	Image-to-Text			Text-to-Image			rSum
	R@1	R@5	R@10	R@1	R@5	R@10	
Single Client	76.6	95.3	98.2	63.0	90.1	95.9	519.1
Federated Model	**80.2**	**96.6**	**98.8**	**64.2**	**91.2**	**96.3**	**527.2**

5. Conclusion

In this paper, we introduced a federated learning framework into bimodal graph neural networks for text-image retrieval, enabling effective training under distributed data storage constraints. Experimental results demonstrate that, compared to conventional centralized approaches, our method significantly improves training efficiency while ensuring data privacy. Moreover, by aggregating knowledge across decentralized clients, the federated models enhance generalization and yield superior retrieval performance. Our findings demonstrate that federated learning can serve as an effective paradigm for scalable and privacy-preserving text-image retrieval. Future work may further investigate the optimization and robustness of bimodal models within federated settings.

Author Contributions: Xueming Yan: Methodology, Investigation, Writing (original draft, review and editing); **Chuyue Wang:** Methodology, Validation, Writing (original draft, review and editing); **Yaochu Jin:** Conceptualization, Writing (review and editing), Supervision, Funding acquisition.

Funding: This work was supported in part by the Guangdong Philosophy and Social Science Foundation (GD25YGG26), the lnternational Collaboration Fund for Creative Research Teams (ICFCRT) of NSFC (W2441019), National Natural Science Foundation of China (62136003), and the Major Program of the Natural Science Foundation of Zhejiang Province (D25F020001).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ebaid, D.B.; Madbouly, M.M.; El-Zoghabi, A.A. Bi-directional image–text matching deep learning-based approaches: Concepts, methodologies, benchmarks and challenges. Int. J. Comput. Intell. Syst., **2023**, *16*: 81. doi:10.1007/s44196-023-00260-3
2. Zhou, Y.H.; Yan, X.M.; Huang, H.; *et al*. Legal text retrieval with contrastive representation learning and evolutionary data augmentation. In *Proceedings of 2024 IEEE Congress on Evolutionary Computation* (*CEC*), *Yokohama, Japan, 30 June 2024–5 July 2024*; IEEE: New York, 2024; pp. 1–7. doi:10.1109/CEC60901.2024.10612052
3. Ren, Z.; Jin, H.L.; Lin, Z.; *et al*. Joint image-text representation by Gaussian visual-semantic embedding. In *Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016*; ACM: New York, 2016; pp. 207–211. doi:10.1145/2964284.2967212
4. Engilberge, M.; Chevallier, L.; Perez, P.; *et al*. Finding beans in burgers: Deep semantic-visual embedding with localization. In *Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018*; IEEE: New York, 2018; pp. 3984–3993. doi:10.1109/CVPR.2018.00419
5. Zhen, L.L.; Hu, P.; Wang, X.; *et al*. Deep supervised cross-modal retrieval. In *Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), *Long Beach, CA, USA, 15–20 June 2019*; IEEE: New York, 2019; 10386–10395. doi:10.1109/CVPR.2019.01064
6. Yan, X.M.; Xue, H.W.; Jiang, S.Y.; *et al*. Multimodal sentiment analysis using multi-tensor fusion network with cross-modal modeling. Appl. Artif. Intell., **2022**, *36*: 2000688. doi:10.1080/08839514.2021.2000688
7. Lee, K.H.; Chen, X.; Hua, G.; *et al*. Stacked cross attention for image-text matching. In *Proceedings of the 15th European Conference on Computer Vision, Munich, Germany, 8–14 September 2018*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 201–216. doi:10.1007/978-3-030-01225-0_13
8. Wang, Y.X.; Yang, H.; Qian, X.M.; *et al*. Position focused attention network for image-text matching. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, Macao, China, 10 August 2019*; AAAI Press: Palo Alto, CA, USA, 2019; pp. 3792–3798.
9. Wei, X.; Zhang, T.Z.; Li, Y.; *et al*. Multi-modality cross attention network for image and sentence matching. In *Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), *Seattle, WA, USA, 13–19 June 2020*; IEEE: New York, NY, USA, 2020; pp. 10938–10947. doi:10.1109/CVPR42600.2020.01095
10. Ji, Z.; Chen, K.X.; Wang, H.R. Step-wise hierarchical alignment network for image-text matching. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence, Montreal, QC, Canada, 9–27 August 2021*; pp. 765–771. doi:10.24963/ijcai.2021/106
11. Yan, X.M.; Huang, H.; Jin, Y.C.; *et al*. Neural architecture search via multi-hashing embedding and graph tensor networks for multilingual text classification. IEEE Trans. Emerg. Top. Comput. Intell., **2024**, *8*: 350–363. doi:10.1109/TETCI.2023.3301774

12. Wang, S.J.; Wang, R.P.; Yao, Z.W.; *et al*. Cross-modal scene graph matching for relationship-aware image-text retrieval. In *Proceedings of 2020 IEEE Winter Conference on Applications of Computer Vision* (*WACV*), *Snowmass, CO, USA, 1–5 March 2020*; IEEE: New York, NY, USA, 2020; pp. 1497–1506. doi:10.1109/WACV45572.2020.9093614
13. Nguyen, M.D.; Nguyen, B.T.; Gurrin, C. A deep local and global scene-graph matching for image-text retrieval. In *New Trends in Intelligent Software Methodologies, Tools and Techniques*; Fujita, H., Perez-Meana, H., Eds.; IOS Press: Amsterdam, The Netherlands, 2021. doi:10.3233/FAIA210049
14. Liu, C.X.; Mao, Z.D.; Zhang, T.Z.; *et al*. Graph structured network for image-text matching. In *Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), *Seattle, WA, USA, 13–19 June 2020*; IEEE: New York, NY, USA, 2020; pp. 10918–10927. doi:10.1109/CVPR42600.2020.01093
15. Diao, H.W.; Zhang, Y.; Ma, L.; *et al*. Similarity reasoning and filtration for image-text matching. In *Proceedings of the 35th AAAIConference on Artificial Intelligence, New York, NY, USA, 2–9 February 2021*; AAAI Press: Palo Alto, CA, USA, 2021; pp. 1218–1226. doi:10.1609/aaai.v35i2.16209
16. Yang, Q.; Liu, Y.; Chen, T.J.; *et al*. Federated machine learning: Concept and applications. ACM Trans. Intell. Syst. Technol. (TIST), **2019**, *10*: 12. doi:10.1145/3298981
17. Liu, D.B.; Miller, T. Federated pretraining and fine tuning of BERT using clinical notes from multiple silos. arXiv, **2020**, arXiv: 2002.08562.
18. Zhuo, Y.X.; Li, B.X. Fedns: Improving federated learning for collaborative image classification on mobile clients. In *Proceedings of 2021 IEEE International Conference on Multimedia and Expo* (*ICME*), *Shenzhen, China, 5–9 July 2021*; IEEE: New York, NY, USA, 2021; pp. 1–6. doi:10.1109/ICME51207.2021.9428075
19. Wang, H.; Zeng, Z.R.; Liu, R.F.; *et al*. A federated learning based Chinese text classification model with parameter factorization weighting. In *Proceedings of the 2021 7th IEEE International Conference on Network Intelligence and Digital Content (IC-NIDC), Beijing, China, 17–19 November 2021*; IEEE: New York, NY, USA, 2021; pp. 299–303. doi:10.1109/IC-NIDC54101.2021.9660471
20. Lyu, L.J.; Yu, H.; Ma, X.J.; *et al*. Privacy and robustness in federated learning: Attacks and defenses. IEEE Trans. Neural Netw. Learn. Syst., 2024, *35*, 8726–8746. doi:10.1109/TNNLS.2022.3216981
21. Faghri, F.; Fleet, D.J.; Kiros, J.R.; *et al*. VSE++: Improved visual-semantic embeddings. arXiv, **2018**, arXiv: 1707.05612.
22. Li, K.P.; Zhang, Y.L.; Li, K.; *et al*. Visual semantic reasoning for image-text matching. In *Proceedings of 2019 IEEE/CVF International Conference on Computer Vision* (*ICCV*), *Seoul, Korea* (*South*), *27 October 2019–2 November 2019*; IEEE: New York, NY, USA, 2019; pp. 4653–4661. doi:10.1109/ICCV.2019.00475
23. Zong, L.L.; Xie, Q.J.; Zhou, J.H.; *et al*. FedCMR: Federated cross-modal retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Online, 11–15 July 2021*; ACM: New York, 2021; pp. 1672–1676. doi:10.1145/3404835.3462989
24. McMahan, B.; Moore, E.; Ramage, D.; *et al*. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017*; pp. 1273–1282.
25. Li, T.; Sahu, A.K.; Zaheer, M.; *et al*. Federated optimization in heterogeneous networks. In *Proceedings of the 3rd Conference on Machine Learning and Systems, Austin, TX, USA, 2–4 March 2020*; pp. 429–450.
26. Ren, S.Q.; He, K.M.; Girshick, R.; *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proceedings of the 29th International Conference on Neural Information Processing Systems, Montreal, Canada, 7–12 December 2015*; MIT Press: Cambridge, UK, 2015; pp. 91–99.
27. Pennington, J.; Socher, R.; Manning, C.D. GloVe: Global vectors for word representation. In *Proceedings of 2014 Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), *Doha, Qatar, 25–29 October 2014*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2014; pp. 1532–1543. doi:10.3115/v1/D14-1162
28. Lin, T.Y.; Maire, M.; Belongie, S.; *et al*. Microsoft COCO: Common objects in context. In *Proceedings of the 13th European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755. doi:10.1007/978-3-319-10602-1_48
29. Li, Q.B.; Diao, Y.Q.; Chen, Q.; *et al*. Federated learning on non-IID data silos: An experimental study. In *Proceedings of the 2022 IEEE 38th International Conference on Data Engineering* (*ICDE*), *Kuala Lumpur, Malaysia, 9–12 May 2022*; IEEE: New York, NY, USA, 2022; pp. 965–978. doi:10.1109/ICDE53745.2022.00077
30. Li, A.; Sun, J.W.; Wang, B.H.; *et al*. LotteryFL: Empower edge intelligence with personalized and communication-efficient federated learning. In *Proceedings of 2021 IEEE/ACM Symposium on Edge Computing* (*SEC*), *San Jose, CA, USA, 14–17 December 2021*; IEEE: New York, NY, USA, 2021; pp. 68–79. doi:10.1145/3453142.3492909
31. Kairouz, P.; McMahan, H.B.; Avent, B.; *et al*. Advances and open problems in federated learning. Found. Trends® Mach. Learn., **2021**, *14*: 1–210. doi:10.1561/2200000083

Citation: Yan, X; Wang, C.; Jin, Y.. Federated Bimodal Graph Neural Networks for Text-Image Retrieval. *International Journal of Network Dynamics and Intelligence*. 2025, 4(2), 100009. doi: 10.53941/ijndi.2025.100009