# Federated Graph Learning - A Position Paper

**Huanding Zhang**[*1], **Tao Shen**[*3], **Fei Wu**[3], **Mingyang Yin**[4], **Hongxia Yang**[4] and **Chao Wu**[†2]
[1]School of Software Technology, Zhejiang University, Hangzhou, China
[2]School of Public Affairs, Zhejiang University, Hangzhou, China
[3]Department of Computer Science, Zhejiang University, Hangzhou, China
[4]DAMO Academy, Alibaba Group, Hangzhou, China
{zhanghuanding, tao.shen, wufei, chao.wu}@zju.edu.cn, {hengyang.ymy, yang.yhx}@alibaba-inc.com

## Abstract

Graph neural networks (GNN) have been successful in many fields, and derived various researches and applications in real industries. However, in some privacy sensitive scenarios (like finance, healthcare), training a GNN model centrally faces challenges due to the distributed data silos. Federated learning (FL) is a an emerging technique that can collaboratively train a shared model while keeping the data decentralized, which is a rational solution for distributed GNN training. We term it as federated graph learning (FGL). Although FGL has received increasing attention recently, the definition and challenges of FGL is still up in the air. In this position paper, we present a categorization to clarify it. Considering how graph data are distributed among clients, we propose four types of FGL: inter-graph FL, intra-graph FL and graph-structured FL, where intra-graph is further divided into horizontal and vertical FGL. For each type of FGL, we make a detailed discussion about the formulation and applications, and propose some potential challenges.

## 1 Motivation

Graph neural networks (GNN) have demonstrated remarkable performance in modeling graph data, and derived various researches and applications in real industries like finance [Liu *et al.*, 2018] [Liu *et al.*, 2019] [Wang *et al.*, 2019], traffic [Yu *et al.*, 2017], recommender systems [Ying *et al.*, 2018], chemistry [Wang *et al.*, 2020b], etc. However, GNN still faces many problems and one of them is the data silos. Because of the privacy concern or commercial competition, data exist in a isolated manner, giving rise to challenges on centrally training GNN. For example, banks may leverage GNN as anti-fraud models, but they only have transactions data of locally registered users (subgraph), thus the model is not effective for other users. Also, pharmaceutical companies usually utilize GNN for drug discovery and synthesis, while the data are quite limited and confidential in independent research institution of companies. Whereas GNN has been successful in many fields, isolated data restrict its further development.

Federated learning (FL) is a machine learning setting where clients can collaboratively train a shared model under the orchestration of central server, while keeping the data decentralized. Unlike traditional centralized machine learning techniques, data are fixed locally, rather than being gathered in central server, who exists many of the systemic privacy risks and costs [Kairouz *et al.*, 2019]. Back to aforementioned examples, with federated learning, banks or pharmaceutical companies can collaboratively train a shared GNN model, utilize isolated data while keeping them safe and local. Hence, FL is a promising solution for training GNN over isolated graph data, and in this paper we term it as federated graph learning (FGL).

As far as we know, FGL has received increasing attention recently. [Zheng *et al.*, 2020] devises a novel FL framework for GNN that supports automatically hyper-parameters optimization. [Wang *et al.*, 2020a] proposes a FL framework for semi-supervised node classification based on meta learning. [Jiang *et al.*, 2020] presents a method to learn dynamic representation of objects from multi-user graph sequences. [Wu *et al.*, 2021] designs a federated GNN framework for privacy preserving recommendation. [Scardapane *et al.*, 2020] presents a distributed training method for GNN, but it preserves the edges among subgraphs.

However, the definition and challenges of FGL is still up in the air. Although [He *et al.*, 2021] proposes a rather comprehensive benchmark for FGL, it is not detailed enough about categorization. In this position paper, we present a categorization to clarify it. Considering how graph data are distributed among clients, we propose four types of FGL: inter-graph FL, intra-graph FL and graph-structured FL, where intra-graph is further divided into horizontal and vertical FGL, referring to the categorization of FL [Yang *et al.*, 2019]. For each type of FGL, we discuss the formulation, applications and challenges. The rest of this paper is organized as follows: In Section 2, we detail four types of FGL. In Section 3, we analysis potential challenges and possible solution for each type of FGL.

*Equal contributions
†Contact Author

## 2 A categorization of federated graph learning

We introduce four types of FGL from the perspective of how graph data are distributed in FL. They are summarized in table 1, details will be discussed as follows. Without loss of generality, we follow the settings of Graph Convolutional Networks (GCN) [Kipf and Welling, 2016] and Federated Averaging (FedAvg) [McMahan *et al.*, 2017] for convenience.

A typical FL framework consists of a server and $K$ clients. The $k^{th}$ client has its own dataset $D_k$ with size of $|D_k| = N_k$, and $N = \sum_{k=1}^{K} N_k$. Graph convolution variants [Veličković *et al.*, 2017] [Hamilton *et al.*, 2017] can be generally formulated as the MPNN framework:

$$x_i^l = \gamma^l(x_i^{l-1}, Aggr^l_{j\in\mathcal{N}_i}\phi^l(x_i^{l-1}, x_j^{l-1}, a_{ij})), \tag{1}$$

where the graph is $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $x_i^l$ is the $i^{th}$ node feature in $l^{th}$ layer, $a_{ij}$ is the edge feature between node $i$ and node $j$, $\mathcal{N}$ denotes the neighbor set of node $i$, $Aggr$ denotes differentiable aggregation function($sum$, $mean$, $max$, etc.), $\gamma$ and $\phi$ denote differentiable function (e.g. MLP). For simplicity, a GCN model composed by 1 can be denoted as $H(X, A, W)$, where $X$ is feature matrix, $A$ is adjacency matrix, and $W$ denotes parameters.

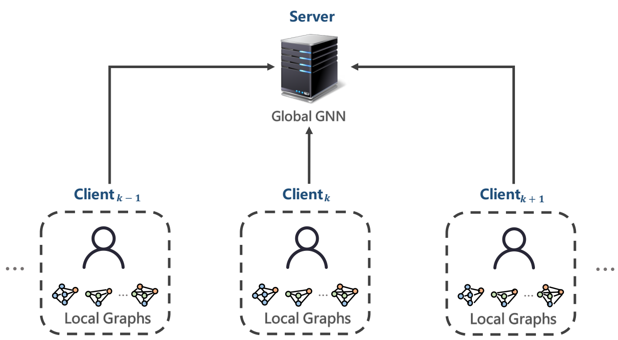


Figure 1: Framework of inter-graph FL: the sample granularity is graph and global GNN model performs graph-level task.

### 2.1 Inter-graph federated learning

This type of FGL is the most natural derivation of FL, where each sample of clients is of graph data, and global model performs graph-level task (shown as figure 1). The most typical application of inter-graph FL is in the biochemical industry where researchers use GNN to study the graph structure of molecules. A molecule can be represented as a graph where atoms are nodes and chemical bonds are edges. In the study of drug properties, every pharmaceutical company holds a confidential dataset $D_k$ which contains molecule structure $\{\mathcal{G}_i\}$ and corresponding properties $\{y_i\}$. In the past, commercial competition hindered their cooperation, but it becomes possible with the framework of inter-graph FL. Under this setting, $D_k = \{(\mathcal{G}_i^{(k)}, y_i^{(k)})\}$, global model is

$$\hat{y}_i^{(k)} = H(X_i^{(k)}, A_i^{(k)}, W), \tag{2}$$

where $X_i^{(k)}$ and $A_i^{(k)}$ denote feature and adjacency matrix of $i^{th}$ graph in $k^{th}$ client's dataset, $\hat{y}$ is output.

Applying FedAvg, the objective function is

$$\begin{aligned} &\min_W \frac{N_k}{N}\sum_{k=1}^{K} f_k(W), \\ &f_k(W) = \frac{1}{N_k}\sum_{i=1}^{N_k}\mathcal{L}(H(X_i^{(k)}, A_i^{(k)}, W), y_i^{(k)}), \end{aligned} \tag{3}$$

where $f_k(W)$ denotes local objective function and $\mathcal{L}$ is global loss. Pharmaceutical companies thus can collaboratively train a shared model without providing confidential data.

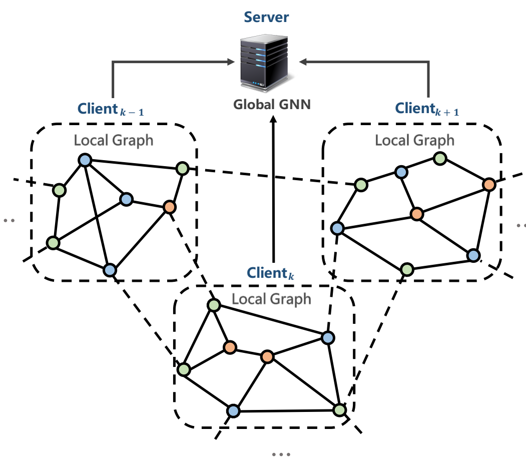


Figure 2: Framework of horizontal intra-graph FL: subgraphs held in clients are horizontally distributed, edges represented as dashed line are the connections that should have been there but are missing.

### 2.2 Intra-graph federated learning

Another type of FGL is the intra-graph federated learning, where each client own a part of latent entire graph. Referring to [Yang *et al.*, 2019], intra-graph federated learning can also be divided into horizontal and vertical FGL, corresponding to users and features who is partitioned.

#### Horizontal intra-graph FL

In this situation, the subgraphs held in each client appear to be horizontally partitioned from the latent entire graph(shown as in figure 2, connections among them are lost because of data isolated storage, strictly speaking, there can be overlap), that is, $A \Rightarrow \{A^{(k)}\}$. Horizontally distributed subgraphs have the same properties, clients share the same feature and label space but different node ID space. Under this setting, $D_k = (\mathcal{G}^{(k)}, Y^{(k)})$, $N_k$ denotes the number of nodes in $\mathcal{G}^{(k)}$. Global GNN model performs node or link-level task,

$$\hat{Y}^{(k)} = H(X^{(k)}, A^{(k)}, W). \tag{4}$$

The objective function becomes

$$\begin{aligned} &\min_{W} \frac{N_k}{N} \sum_{k=1}^{K} f_k(W), \\ f_k(W) =& \mathcal{L}(H(X^{(k)}, A^{(k)}, W), Y^{(k)}). \end{aligned} \tag{5}$$

Subgraph horizontal distribution is very common in real world. For example, in online social app, each user has a local social network $\mathcal{G}^{(k)}$ and $\{\mathcal{G}^{(k)}\}$ constitute the latent entire human social network $\mathcal{G}$. The developers are able to devise friend recommendation algorithm based on horizontal intra-graph FL to avoid violating users' social privacy.

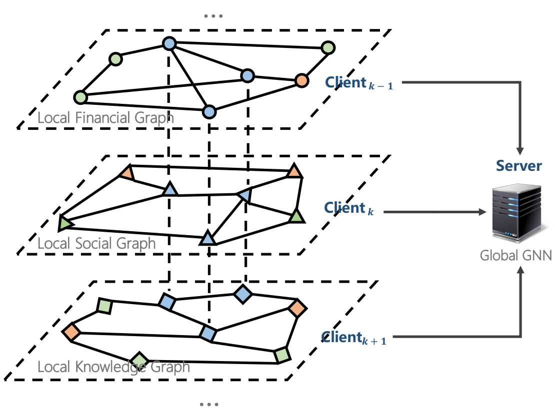


Figure 3: Framework of vertical intra-graph FL: subgraphs held in clients are vertically distributed, and they are parallel and heavily overlap with each other, vertical dashed lines indicate the corresponding nodes have same ID.

### Vertical intra-graph FL

Subgraph vertical distribution means that they are parallel and heavily overlap with each other(shown as figure 3, graphs of financial, social and knowledge are vertically distributed). It's like the latent entire graph is vertically partitioned, that is, $A \Rightarrow \{A^{(k)}\}$, $X \Rightarrow \{X^{(k)}\}$, $Y \Rightarrow \{Y^{(k)}\}$. Under this setting, clients share the same node ID space but different feature and label space, $D_k = (\mathcal{G}^{(k)}, Y^{(k)})$, $V'$ is set of common nodes, $N_k$ is size of $V'$. Global model is not unique(it depends on how many clients have labels), which indicates vertical intra-graph FL supports multi-task learning. The main purpose of vertical intra-graph FL is to learn more comprehensive GNN by combining $\{X_v^{(k)} | v \in V'\}$ and sharing $\{Y_v^{(k)} | v \in V'\}$ in a privacy preserved and communication efficient manner. Without considering the method of entity matching and data sharing, the objective function can be expressed as

$$\begin{aligned} &\min_{W} f_k(W), \\ f_k(W) = \mathcal{L}(H(&Aggr_{k=1}^{K}(X_{V'}^{(k)}), A_{V'}^{(k)}, W^{(k)}), Y_{V'}^{(k)}). \end{aligned} \tag{6}$$

Vertical intra-graph FL can be applied in the cooperation among organizations. For example, in detection of money laundering, criminals are tends to devise sophisticated strategies that span across different organizations. Due to privacy concern, banks need to hand over list of suspects to a trustworthy national institution and rely on them to do analysis. This procedure is inefficient. With the framework of vertical intra-graph FL, banks are able to collaboratively monitor money laundering activities in real-time while keeping their users' data protected. Some researchers have studied this type of FGL, [Suzumura *et al.*, 2019] and [Chen *et al.*, 2020] respectively devise a vertical intra-graph FL framework for financial fraud detection and knowledge graph embedding.

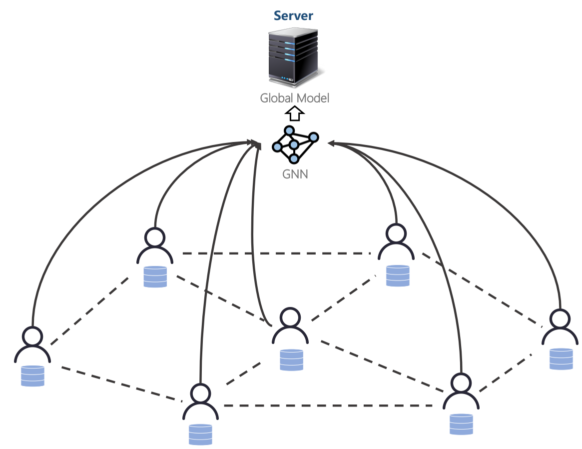


Figure 4: Framework of graph-structured FL: graphs exist as relationships among clients, GNN is used to extract inherent information from the topology of clients.

## 2.3 Graph-structured federated learning

In addition to being data, graphs can also exit as relationships among clients (e.g. geography or social networks shown as figure 4), that is, $client^{(k)} \Rightarrow v_k$. Graph-structured federated learning refers to the situation where server uses GNN to aggregate local models based on clients topology.

$$f(w)^{(t)} = Aggr_{k=1}^{K}(f(w)^{(k,t-1)}, A, W). \tag{7}$$

Under this setting, clients hold either Euclidean or graph data, global model performs any kind of task, objective function is the same as in FL. Graph-structured FL can be considered as a special federated optimization method since GNN is applied to extract inherent information among clients to improve FL. The typical application is in federated traffic flow prediction where monitoring devices are distributed in different geographic positions, GNN are used to capture spatial dependencies among devices [Meng *et al.*, 2021].

# 3 Challenges

Though researchers have already proposed several FGL frameworks, there are still many problems. Most of them are left from FL and become more complicated in graph domain, such as Non-IID data, communication efficiency and robustness [Kairouz *et al.*, 2019]. In this section, we discuss the main challenges and possible solutions for each type of FGL.

| Type | | Federalization | Data form in clients | Global model task |
|---|---|---|---|---|
| Inter-graph Federated Learning | | $D_k = \{\mathcal{G}_i\}$ | Graphs | Graph-level |
| Intra-graph federated learning | Horizontal | $A \Rightarrow \{A^{(k)}\}$ | Horizontally distributed subgraphs | Node or link-level |
| | Vertical | $A \Rightarrow \{A^{(k)}\}$<br>$X \Rightarrow \{X^{(k)}\}$<br>$Y \Rightarrow \{Y^{(k)}\}$ | Vertically distributed subgraphs | Node or link-level |
| Graph-structured federated learning | | $client^{(k)} \Rightarrow v_k$ | Arbitrary | Arbitrary |

Table 1: Four types of FGL: each type corresponds to a different way of federalization of graph.

### 3.1 Non-IID graph structure

Non-IID problem is inevitable no matter in which type of FGL. Same as in FL, it can both impact convergence speed and accuracy. Researchers have attempted to devise some methods to alleviate its influence [Zheng *et al.*, 2020] [Wang *et al.*, 2020a], as far as we know, there is no work solving it completely. In addition to feature and label, graph data have edge (structure) information, which indicates Non-IID of graph structure might influence the learning process as well. Properties of graph structure include degree distribution, average path length, average clustering coefficient, etc. Studying Non-IID of these properties might be an important aspect of solving Non-IID problem in graph domain. No work has paid attention to studying Non-IID of graph structure yet, and it's worth digging.

### 3.2 Isolated graph in horizontal intra-graph FL

Representation learning on graph models relies on walking or message passing through multi-order neighbors. However, the latent entire graph is isolated by different data holders in horizontal intra-graph FL, the diameter of local subgraph is nearly small. It will impact the accuracy of GCN since the local graph cannot provide information from high-order neighbors. Consider an extreme case where the local subgraph only contains one node, GCN degenerates to MLP. Therefore, discovering latent edges among local subgraphs of clients is a crucial challenge in horizontal intra-graph FL, there are some researches who have mentioned it, [Wu *et al.*, 2021] proposes a method based on homomorphic encryption to expand local subgraph, more ideas are needed on this issue.

### 3.3 Entities matching and secure data sharing in vertical intra-graph FL

Entities matching and secure data sharing are key problems for both vertical FL and vertical intra-graph FL. [Hardy *et al.*, 2017] achieves learning a federated logistic regression model between two participants based on additively homomorphic encryption, and [Feng and Yu, 2020] generalizes it to multi-participants and multi-class classification. Vertical intra-graph FL is at least as complicated as VFL, and the main difficulties also lie in ensuring precision, privacy preserving and communication efficient at the same time. There is no vertical intra-graph FL framework achieving these requires. [Chen *et al.*, 2020] proposes a federated framework to do knowledge graph embedding by a matching table held in server, which violates privacy preserving to some extent.

### 3.4 Dataset of intra-graph FL

The richness of image and corpus dataset is a necessary condition for rapid development of computer vision and natural language processing. However, there is no suitable graph dataset for intra-graph FL. For Euclidean data, we can easily simulate data distribution in experiments. However, simulation becomes difficult when it comes to graph data due to the additional structure information. For example, in horizontal setting, we have to split a graph into multiple subgraphs but the removed edges and subgraph distribution are not in line with reality. It can be hard in vertical setting as well. Although features can be split into several partitions, whether all partitions have the same structure needs to be considered. It is usually more complicated in real scenes. Thus, the lack of datasets limits the development of intra-graph FL.

### 3.5 Communication and memory consumption

Communication and memory consumption turns out to be a key bottleneck when applying federated algorithms in reality. For example, for federated recommender system, models transmitted between server and clients may be heavy, where user/item representation layers occupy most of model parameters and the size of representation parameters grows linearly with the ever-increasing scale of user/item. It brings unfavorable both communication and memory consumption. Model quantization, pruning, distillation are effective methods for model compression. [Tailor *et al.*, 2020] studies model quantization method for GNN. [Yang *et al.*, 2020] proposes a distillation approach which transfers topology-aware knowledge from teacher GCN to student GCN. [Lian *et al.*, 2020] devises an end-to-end framework for learning quantization of item representation in recommender system. Thus, compression technique for GNN is also a potential way for FGL.

## References

[Chen *et al.*, 2020] Mingyang Chen, Wen Zhang, Zonggang Yuan, Yantao Jia, and Huajun Chen. Fede: Embedding knowledge graphs in federated setting. *arXiv preprint arXiv:2010.12882*, 2020.

[Feng and Yu, 2020] Siwei Feng and Han Yu. Multi-participant multi-class vertical federated learning. *arXiv preprint arXiv:2001.11154*, 2020.

[Hamilton *et al.*, 2017] William L Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. *arXiv preprint arXiv:1706.02216*, 2017.

[Hardy *et al.*, 2017] Stephen Hardy, Wilko Henecka, Hamish Ivey-Law, Richard Nock, Giorgio Patrini, Guillaume Smith, and Brian Thorne. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *arXiv preprint arXiv:1711.10677*, 2017.

[He *et al.*, 2021] Chaoyang He, Keshav Balasubramanian, Emir Ceyani, Yu Rong, Peilin Zhao, Junzhou Huang, Murali Annavaram, and Salman Avestimehr. Fedgraphnn: A federated learning system and benchmark for graph neural networks. *arXiv preprint arXiv:2104.07145*, 2021.

[Jiang *et al.*, 2020] Meng Jiang, Taeho Jung, Ryan Karl, and Tong Zhao. Federated dynamic gnn with secure aggregation. *arXiv preprint arXiv:2009.07351*, 2020.

[Kairouz *et al.*, 2019] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

[Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[Lian *et al.*, 2020] Defu Lian, Haoyu Wang, Zheng Liu, Jianxun Lian, Enhong Chen, and Xing Xie. Lightrec: A memory and search-efficient recommender system. In *Proceedings of The Web Conference 2020*, pages 695–705, 2020.

[Liu *et al.*, 2018] Ziqi Liu, Chaochao Chen, Xinxing Yang, Jun Zhou, Xiaolong Li, and Le Song. Heterogeneous graph neural networks for malicious account detection. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 2077–2085, 2018.

[Liu *et al.*, 2019] Ziqi Liu, Chaochao Chen, Longfei Li, Jun Zhou, Xiaolong Li, Le Song, and Yuan Qi. Geniepath: Graph neural networks with adaptive receptive paths. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4424–4431, 2019.

[McMahan *et al.*, 2017] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[Meng *et al.*, 2021] Chuizheng Meng, Sirisha Rambhatla, and Yan Liu. Cross-node federated graph neural network for spatio-temporal data modeling, 2021.

[Scardapane *et al.*, 2020] Simone Scardapane, Indro Spinelli, and Paolo Di Lorenzo. Distributed graph convolutional networks. *arXiv preprint arXiv:2007.06281*, 2020.

[Suzumura *et al.*, 2019] Toyotaro Suzumura, Yi Zhou, Natahalie Baracaldo, Guangnan Ye, Keith Houck, Ryo Kawahara, Ali Anwar, Lucia Larise Stavarache, Yuji Watanabe, Pablo Loyola, et al. Towards federated graph learning for collaborative financial crimes detection. *arXiv preprint arXiv:1909.12946*, 2019.

[Tailor *et al.*, 2020] Shyam A Tailor, Javier Fernandez-Marques, and Nicholas D Lane. Degree-quant: Quantization-aware training for graph neural networks. *arXiv preprint arXiv:2008.05000*, 2020.

[Veličković *et al.*, 2017] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[Wang *et al.*, 2019] Daixin Wang, Jianbin Lin, Peng Cui, Quanhui Jia, Zhen Wang, Yanming Fang, Quan Yu, Jun Zhou, Shuang Yang, and Yuan Qi. A semi-supervised graph attentive network for financial fraud detection. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 598–607. IEEE, 2019.

[Wang *et al.*, 2020a] Binghui Wang, Ang Li, Hai Li, and Yiran Chen. Graphfl: A federated learning framework for semi-supervised node classification on graphs. *arXiv preprint arXiv:2012.04187*, 2020.

[Wang *et al.*, 2020b] Hanchen Wang, Defu Lian, Ying Zhang, Lu Qin, and Xuemin Lin. Gognn: Graph of graphs neural network for predicting structured entity interactions. *arXiv preprint arXiv:2005.05537*, 2020.

[Wu *et al.*, 2021] Chuhan Wu, Fangzhao Wu, Yang Cao, Yongfeng Huang, and Xing Xie. Fedgnn: Federated graph neural network for privacy-preserving recommendation. *arXiv preprint arXiv:2102.04925*, 2021.

[Yang *et al.*, 2019] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

[Yang *et al.*, 2020] Yiding Yang, Jiayan Qiu, Mingli Song, Dacheng Tao, and Xinchao Wang. Distilling knowledge from graph convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7074–7083, 2020.

[Ying *et al.*, 2018] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 974–983, 2018.

[Yu *et al.*, 2017] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

[Zheng *et al.*, 2020] Longfei Zheng, Jun Zhou, Chaochao Chen, Bingzhe Wu, Li Wang, and Benyu Zhang. Asfgnn: Automated separated-federated graph neural network. *arXiv preprint arXiv:2011.03248*, 2020.