Data Pricing for Graph Neural Networks without Pre-purchased Inspection

Yiping Liu*
UESTC, China
The University of Auckland, New Zealand
yliu823@aucklanduni.ac.nz

Mengxiao Zhang†*
UESTC, China
The University of Auckland, New Zealand
mengxiao.zhang@auckland.ac.nz

Jiamou Liu
The University of Auckland, New Zealand
jiamou.liu@auckland.ac.nz

Song Yang
The University of Auckland, New Zealand
syan382@aucklanduni.ac.nz

ABSTRACT

Machine learning (ML) models have become essential tools in various scenarios. Their effectiveness, however, hinges on a substantial volume of data for satisfactory performance. Model marketplaces have thus emerged as crucial platforms bridging model consumers seeking ML solutions and data owners possessing valuable data. These marketplaces leverage model trading mechanisms to properly incentive data owners to contribute their data, and return a well performing ML model to the model consumers. However, existing model trading mechanisms often assume the data owners are willing to share their data before being paid, which is not reasonable in real world. Given that, we propose a novel mechanism, named Structural Importance based Model Trading (SIMT) mechanism, that assesses the data importance and compensates data owners accordingly without disclosing the data. Specifically, SIMT procures feature and label data from data owners according to their structural importance, and then trains a graph neural network for model consumers. Theoretically, SIMT ensures incentive compatible, individual rational and budget feasible. The experiments on five popular datasets validate that SIMT consistently outperforms vanilla baselines by up to 40% in both MacroF1 and MicroF1.

KEYWORDS

Model marketplaces; data pricing; structural entropy

ACM Reference Format:
Yiping Liu, Mengxiao Zhang, Jiamou Liu, and Song Yang. 2025. Data Pricing for Graph Neural Networks without Pre-purchased Inspection. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 9 pages.

1 INTRODUCTION

In today's digital age, data has become an essential asset, serving as the foundation for AI and machine learning advancements. To meet the increasing demand for high-quality data, a new business paradigm known as the *model marketplace* has emerged [22], exemplified by platforms like Modzy. A model marketplace facilitates the exchange between *model consumers*, who seek AI models for various tasks, and *data owners*, who possess the feature and label data necessary for model training. The marketplace purchases data from data owners, uses it to train AI models, and then sells these trained models to consumers. However, a key challenge in model marketplaces is determining how to properly compensate data owners for their contributions, a problem referred to as *data pricing*. This problem is challenging because the importance of data is difficult to evaluate. Most existing studies assume that marketplaces acquire data from data owners *before* paying them and use the subsequent performance improvements as a measure of data importance. For example, [2, 11, 16, 22, 42] rely on this assumption to establish pricing mechanisms based on the marginal impact of data on model accuracy. However, this pre-purchased inspection assumption is impractical in real-world settings. Data owners are often unwilling to release their data without proper payment, fearing that the data, once disclosed, may immediately provide valuable insights to buyers, reducing the incentive to pay.

This leads to a critical question: *How can we measure data importance for model training without direct inspection, thereby facilitating data pricing?* Several studies have attempted to address this question by introducing exogenous metrics for measuring data importance, such as data age, accuracy, volume [13], the extent of perturbations [9, 37], or data owners' reputation [44]. However, these metrics often fail to accurately reflect the contribution of data in the context of model training, particularly when dealing with complex models like Graph Neural Networks (GNNs).

Graph-structured data is prevalent in many real-world scenarios, where the relationships between entities are often as important as their attributes. GNNs excel in tasks involving such data, capturing both node features and network structure. However, data ownership is often decentralized, with different entities controlling separate "pockets" of the network. This creates a need for a marketplace where subgraphs can be purchased and integrated to enable comprehensive model training [3]. For example, in finance, each bank holds its own subset of transaction data, but detecting fraud often requires analysing transaction flows across multiple institutions. Similarly, in healthcare, patient interactions are fragmented across hospitals, clinics, and insurance companies, forming an interconnected yet distributed network. In supply chain management, companies typically have visibility into their direct suppliers

*Equal contribution
†Corresponding authors

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA.

and customers, but the complete supply chain network spans many interdependent organisations. In all of these cases, the full value of the network data cannot be realised without aggregating subgraphs from multiple sources. For data consumers, aggregating subgraphs from multiple sources is essential for training reliable GNN models, enabling applications like fraud detection, personalized healthcare, and supply chain risk analysis.

In this paper, we advance existing research by exploring the pricing of individual data points within subgraphs for GNN training. This introduces a distinct challenge, as the value of any given node to the model's performance is highly dependent on its structural role and connectivity within the broader network. To address this, we aim to develop pricing mechanisms that capture the marginal contribution of each data point, taking into account both its local features and its position within the global network structure. Notably, this is the first work to tackle the problem of pricing graph-structured data in a model marketplace for GNNs.

In the following, we list our main contributions:

- We propose *Structural Importance based Model Trading* (SIMT), a novel model marketplace framework for GNNs that integrates two phases: *data procurement* and *model training*. Figure 1 shows the conceptual framework of SIMT.
- For data procurement, we put forward a new method for assessing the importance of graph-structured data. For this we present a novel *marginal structural entropy* to quantify node informativeness. This method of importance assessment is integrated with an auction mechanism to select data owners and fairly compensate them based on their contributions. We prove that this mechanism is incentive compatible, individual rational, and budget feasible.
- For model training, we introduce the method of *feature propagation* to impute missing feature data for unselected nodes, enabling effective learning with partial data. We also design an *edge augmentation* method to enhance graph structure by adding connections involving unselected nodes, improving the GNN's ability to generalize.
- The proposed SIMT method was evaluated on five well-established benchmark datasets, and consistently outperformed four baseline mechanisms in node classification tasks. SIMT achieved up to a 40% improvement in MacroF1 and MicroF1 scores compared to the Greedy and ASCV methods, demonstrating its superior performance under various budget constraints.

2 RELATED WORK

Data pricing has been extensively studied in two main contexts: *data marketplaces* and *model marketplaces*.

Data Pricing in Data Marketplaces. In data marketplaces, pricing mechanisms revolve around trading raw datasets or simple queries. Previous work has focused on pre-purchase decisions, where data is evaluated before it is accessed, which aligns with our setting. For instance, the importance of datasets is often quantified by metrics such as size, as explored by [19], or privacy levels, as in [28]. Other studies, such as [41] and [15], assess data importance based on its utility to consumers, proposing auction mechanisms and contracts to compensate data owners accordingly.

When it comes to *query-based data pricing*, metrics like privacy levels directly impact the accuracy of responses, thereby influencing data value. For instance, [12, 31] propose auction mechanisms that incorporate privacy in queries, while [21] introduces a take-it-or-leave-it contract for count queries. Further work by [10, 45] expands these ideas to linear predictor queries and broader query settings.

In data marketplaces, data importance is often easily quantifiable using metrics like size or privacy levels. However, in the context of *model marketplaces*, the contribution of individual data points to machine learning model performance is more complex and requires novel pricing methods.

Data Pricing in Model Marketplaces. In model marketplaces, data pricing is typically based on how much a dataset improves a machine learning model's performance. [1] introduced a theoretical framework for data pricing that balances budget constraints with model performance. Subsequent works, such as those by [2] and [11], assume the model's benefit is known and focus on fairly distributing rewards among data owners. A common method for this is the *Shapley value* [32], which compensates each data owner based on their contribution to the model.

Various studies have refined the utility function used in Shapley value calculations by incorporating additional factors. [11] and [16], for example, include K-nearest neighbors and privacy considerations in their utility designs. [22] builds on this by extending the Shapley value framework to model marketplaces. Other research, such as [34] and [26], explores utility design in *collaborative machine learning* scenarios, where data owners also serve as model consumers. In these cases, utility is defined either as the sum of the model's value to the owner and its marginal value to others [26], or through metrics like information gain [34]. [42] and [14] further define utility based on the cosine similarity of parameters or the privacy leakage of shared model parameters.

A common limitation of these works is that they often require training models on the entire dataset before compensating data owners. In practice, this assumption is often unrealistic, as data owners are usually hesitant to contribute their data upfront without proper guarantees or compensation.

A more realistic setting, which is closer to our approach, has been explored by studies [9, 37, 44]. [9] assume that data importance is known and apply a *VCG auction mechanism* to select and compensate data owners. [44] propose an auction mechanism that incorporates the reputation of data owners as a reflection of their contribution, while [37] design an auction that selects data owners based on their privacy requirements. Although these approaches offer valuable insights, they rely on exogenous metrics, such as reputation or privacy, which are often difficult to obtain or may not accurately reflect the intrinsic value of data for model training.

In contrast, our work proposes a novel method to *measure data importance* without direct data inspection. By focusing on the structural properties of graph data and using techniques like structural entropy, we aim to create a fair and effective data pricing mechanism that overcomes the limitations of previous methods.

Comparison with FL and AL. While *Federated Learning* (FL) and *Active Learning* (AL) are well-known paradigms for training models with distributed data, our approach differs in key ways. **(1)** In FL, each data owner trains a local model on private data, which is then aggregated into a global model while preserving privacy [43]. SIMT, by contrast, does not require data owners to train models. Instead, data is directly provided to a central model,

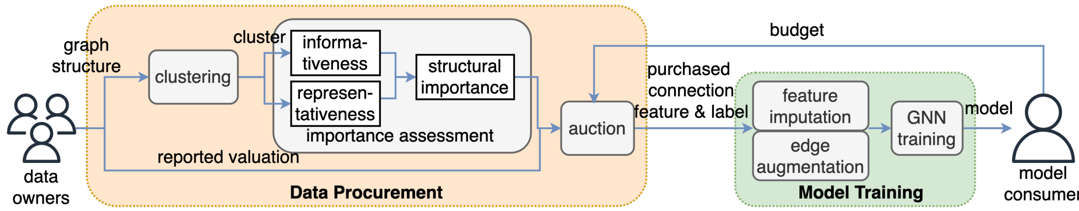

Figure 1: The framework of structural importance-based model trading (SIMT) mechanism.

allowing for optimizations like data augmentation that are not possible in FL's gradient-based aggregation. This eliminates the computational burden on data owners and allows for more flexible model improvements. **(2)** In AL, the model iteratively queries data points to refine learning, typically in multiple rounds [29]. SIMT, however, collects data in a single round, reducing overhead and cost. Furthermore, while AL assumes access to unlabeled data with labels provided iteratively [7, 46], SIMT addresses the real-world challenge of compensating data owners, ensuring they are fairly rewarded for their contributions upfront.

3 PROBLEM FORMULATION

3.1 Model Marketplace for Graph Data

We consider a model marketplace where a *model consumer* interacts with multiple *data owners* to trade graph-structured data. This data is distributed among the various data owners. Let the overall graph be represented as an attributed graph $\mathcal{G} := (V, E, \boldsymbol{X}, \boldsymbol{y})$, where V is the set of nodes, representing individual data subjects, $E \subseteq V \times V$ is the set of edges. $\boldsymbol{X} \in \mathbb{R}^{n \times m}$ is the feature matrix, where n is the number of nodes and m is the dimensionality of the feature vector, and $\boldsymbol{y} \in \mathbb{R}^n$ is the label vector, where each entry corresponds to a label for each node. The adjacency matrix and normalised adjacency matrix of $\mathcal{G}$ are denoted as $\boldsymbol{A}$ and $\tilde{\boldsymbol{A}}$, resp.

For each node $v \in V$, let $\boldsymbol{x}_v \in \mathbb{R}^m$ and y_v represent the feature vector and label of node v, resp. $N_v \subseteq V$ represents the set of neighbours of node v, and $d_v := |N_v|$ denote the degree of node v.

The graph $\mathcal{G}$ is distributed among multiple data owners, each controlling a subgraph. Let O denote the set of data owners, where $o := |O|$ represents the total number of data owners. For each data owner $i \in O$, the *subgraph held by owner* i is represented as $\mathcal{G}_i := (V_i, E_i, \boldsymbol{X}[V_i], \boldsymbol{y}[V_i])$, where $V_i \subseteq V$ is the set of nodes controlled by data owner i, $E_i = E \cap (V_i \times V_i)$ is the set of edges between nodes in V_i, $\boldsymbol{X}[V_i]$, and $\boldsymbol{y}[V_i]$ are the feature matrix, and label vector induced by the nodes in V_i, resp. Let $n_i := |V_i|$ be the number of nodes in subgraph $\mathcal{G}_i$.

Denote the edges within subgraphs as $\dot{E}$ and the edges between the subgraphs as $\ddot{E}$. We assume that $\dot{E} \cap \ddot{E} = \emptyset$ and then $E = \dot{E} \cup \ddot{E}$. We also assume that the internal structure of each subgraph, including the features and labels of nodes, is private to the corresponding data owner. However, the connections between subgraphs (i.e., $\ddot{E}$ the edges connecting nodes from different subgraphs) are known by the model consumer. Data owners are willing to sell the feature, label, and connection data for the nodes they control.

Each data owner $i \in O$ attaches a valuation to her attribute and label data of a single node, denoted by $\theta_i \in \Theta$, where Θ is the set of all possible valuations. The valuation θ_i indicates the minimum payment required by the data owner to allow the use of the attribute and connection data of a single node for model training. The valuation θ_i is privately known only to the data owner, but they may report a different valuation $\theta'_i \neq \theta_i$ if it serves their interests. We assume that each data owner values all their data subjects *equally*, implying that the total valuation is linearly dependent on the number of data records. Let $\boldsymbol{\theta}_i$ be i's valuation vector for all nodes, i.e., $\boldsymbol{\theta}_i := (\theta_{i,1}, \ldots, \theta_{i,n_i}) = \theta_i \cdot \mathbf{1}$, where $\theta_{i,v}$ is the valuation of i for node v. The valuation of all data owners form a valuation matrix, denoted by $\boldsymbol{\theta}$, which is the concatenation $\boldsymbol{\theta}_1 \mid \cdots \mid \boldsymbol{\theta}_o \in \Theta^n$. The model consumer has a *budget*, denoted by $\beta \in \mathbb{R}^+$, for buying the prediction model trained on structure and attribute data.

The model marketplace involves designing a mechanism that procures the attribute/structure data from data owners, and train a GNN model for the model consumer.

3.2 Incentive Mechanism

Definition 3.1. A *mechanism* M consists of two functions, $(\pi(\cdot), p(\cdot))$, where $\pi \colon \Theta^n \to \{0,1\}^n$ is an *allocation function* and $p \colon \Theta^n \to \mathbb{R}^n$ is a *payment function.*

Given a set of data owners and a model consumer, the mechanism takes the reported valuation $\boldsymbol{\theta}' \in \Theta^n$ as input, and outputs allocation result and payment result. The allocation function and the payment function determine which nodes are selected for model training and how much to pay for the data owners, resp. We write the allocation result $\boldsymbol{\pi}(\boldsymbol{\theta}')$ as $(\boldsymbol{\pi}_1(\boldsymbol{\theta}'), \ldots, \boldsymbol{\pi}_o(\boldsymbol{\theta}'))$ and the payment result $\boldsymbol{p}(\boldsymbol{\theta}')$ as $(\boldsymbol{p}_1(\boldsymbol{\theta}'), \ldots, \boldsymbol{p}_o(\boldsymbol{\theta}'))$, where each $\boldsymbol{\pi}_i(\boldsymbol{\theta}'), \boldsymbol{p}_i(\boldsymbol{\theta}')$ is a n_i-dimensional vector with each element $\pi_{i,v}, p_{i,v}$ being an allocation and payment for i's node v. The allocation and payment of node v give data owner i a utility $u_{i,v}(\boldsymbol{\theta}') = (p_{i,v}(\boldsymbol{\theta}') - \theta_{i,v})\pi_{i,v}(\boldsymbol{\theta}')$. The utility of data owner i is $u_i = \sum_{v \in V_i} u_{i,v}$. Once a node is selected, its connection, feature and label data are used for model training.

Let $\boldsymbol{\theta}_{-i}$ denote the valuation of all data owner but i and Θ_{-i} denote the set of all possible $\boldsymbol{\theta}_{-i}$. A mechanism M should satisfy:

- *Incentive Compatible* (IC): Each data owner $i \in O$ gains maximum utility when truthfully reporting her valuation, i.e., $u_i(\theta_i, \boldsymbol{\theta}_{-i}) \geq u_i(\theta'_i, \boldsymbol{\theta}_{-i})$, $\forall \theta_i, \theta'_i \in \Theta, \forall \boldsymbol{\theta}_{-i} \in \Theta_{-i}$.
- *Individual Rational* (IR): Each data owner $i \in O$ gains an non-negative utility when participating in the mechanism, i.e., $u_i(\theta_i, \boldsymbol{\theta}_{-i}) \geq 0$, $\forall \theta_i \in \Theta, \forall \boldsymbol{\theta}_{-i} \in \Theta_{-i}$.
- *Budget Feasible* (BF): Total payment given to all data owners is not exceed the budget β, i.e., $\sum_{i \in O} \boldsymbol{p}_i \boldsymbol{\pi}_i \leq \beta$.

3.3 Graph Neural Network Models

We use GNN as the prediction model. Given a graph, GNN predicts the node labels by stacking multiple layers. Let L be the number of layers in a GNN model. The main idea is to iteratively aggregate the feature information of each node from its neighbours. Specifically, given an attributed graph $\mathcal{G} = (V, E, \boldsymbol{X}, \boldsymbol{y})$, and a GNN with L convolution layers, at a layer $\ell \le L$, the feature embedding $\boldsymbol{h}_v^\ell$ of node $v \in V$ is generated through aggregation and update:

- Aggregation: aggregate the feature embeddings $\boldsymbol{h}_u^\ell$ of all neighbours u of v by an aggregate function such as mean and sum, with trainable weights, i.e., $\boldsymbol{n}_v^\ell \coloneqq \text{Aggregator}^\ell\left(\{\boldsymbol{h}_u^\ell, \forall u \in N_v\}\right)$.
- Update: update the feature embedding $\boldsymbol{h}_v^{\ell+1}$ at the next layer by an update function of the embedding $\boldsymbol{h}_v^\ell$ and the aggregated embeddings $\boldsymbol{n}_v^\ell$, i.e., $\boldsymbol{h}_v^{\ell+1} \coloneqq \text{Updater}^\ell\left(\boldsymbol{h}_v^\ell, \boldsymbol{n}_v^\ell\right)$. Initially, the feature embedding of node v is its feature vector, i.e., $\boldsymbol{h}_v^0 \coloneqq \boldsymbol{x}_v$.

3.4 Optimisation Problem

As discussed in the Introduction, a key issue in determining compensation for data owners in a model marketplace is assessing the importance of their data to model training without direct inspection. To summarise, the problem in this paper is:

Given the model marketplace with a model consumer and several data owners, we, as a *data broker*, aim to design a mechanism that procures the attribute/structural data from data owners, and train a GNN model for the consumer with the following subgoals:

- assessing the importance of data to model training without disclosing the feature and label data;
- optimising GNN performance within the budget; and
- ensuring the mechanism is IC, IR, and BF.

More formally, let $\circ$ denote the Hadamard product operator, which selectively includes elements from $\dot{E}, \boldsymbol{X}, \boldsymbol{y}$ according to the indicator vector $\boldsymbol{\pi} = (\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_o)$. We define $f_{GNN}(\cdot)$ as the output of a GNN model trained on a selected subset of the data with the known $\ddot{E}$. The problem in the paper can be formulated as:

$$\begin{aligned}
\min \quad & ||\boldsymbol{y} - f_{GNN}(\boldsymbol{\pi} \circ \mathcal{G})|| \\
\text{s.t.} \quad & u_i(\theta_i, \boldsymbol{\theta}_{-i}) \ge u_i(\theta_i', \boldsymbol{\theta}_{-i}),\ \forall \theta_i, \theta_i' \in \Theta, \forall \boldsymbol{\theta}_{-i} \in \Theta_{-i}. && \text{(IC)} \\
& u_i(\theta_i, \boldsymbol{\theta}_{-i}) \ge 0,\ \forall \theta_i \in \Theta, \forall \boldsymbol{\theta}_{-i} \in \Theta_{-i} && \text{(IR)} \\
& \sum_{i \in O} \boldsymbol{p}_i \boldsymbol{\pi}_i \le \beta && \text{(BF)} \\
& \boldsymbol{p}_i \ge 0, \pi_{i,v} \in \{0, 1\} \quad \forall i \in O, \forall v \in V_i
\end{aligned}$$

4 PROPOSED METHOD

In this section, we propose a mechanism that procures the most contributing data and trains a GNN model using the procured data. By considering the correlation between graph structure, features, and labels, we leverage the graph structure to offer insights into the contribution of the associated data. Then, we combine this contribution assessment with the data owners' valuation in an auction mechanism to select the most cost-effective data. Subsequently, we augment the procured data using feature imputation and edge augmentation and use the augmented data to train a two-layer GNN model, which is returned to the model consumer. The overall framework is shown in Figure 1.

4.1 Structural Importance

We begin by evaluating the importance of data to model training without inspecting the feature and label data. Our solution is motivated by the observation that the structure of a graph often encodes valuable information about its features and labels. According to the well-known homophily assumption, nodes with similar features and labels are more likely to be closely connected [25, 39]. This is further validated by our case studies on five real-world graphs. We analyse the connections both within and between classes, and the results show that the number of edges within the same class is substantially higher than between different classes, as illustrated in Figure 2. The strong correlation between graph structure and the associated features and labels motivates our approach to leverage the graph structure as auxiliary information in the data selection process. Thus we propose to use the *structural importance* of data owner to represent her data importance.

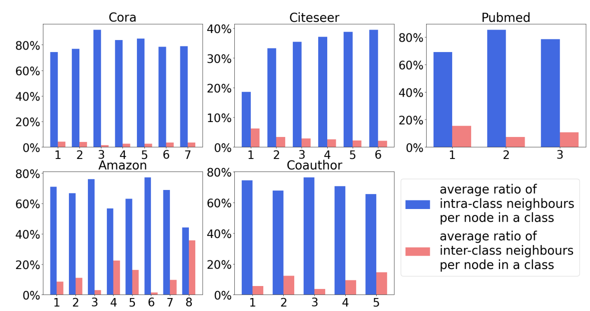

Figure 2: Proportion of intra-class and inter-class edges

The sample dataset for model training should be both informative, reducing the uncertainty of the model, and representative, capturing the overall pattern of the entire dataset [46]. Therefore, we measure the structural importance of data owners in terms of informativeness and representativeness. Nevertheless, determining the informativeness and representativeness of nodes in a graph often relies on the true classes and node features, which are not available due to the absence of true feature and label data. To address this, we first use *structural clusters* to approximate the true classes. With this clustering, we propose the notion of *marginal structural entropy* to quantifies informativeness, and deploy PageRank centrality to quantify representativeness.

Structuring clustering. A crucial tool for the structural clustering is *structural entropy*. Let $G = (V, E)$ represent a graph without attributes. Suppose $P \coloneqq \{C_1, C_2, \ldots, C_T\}$ is a partition of V, where each C_t is called a *cluster* and T is the number of clusters. *Structural entropy* of G relative to P captures the information gain due to the partition. For each cluster $C_t \in P$, write d_t as the sum of degrees d_v of all nodes $v \in C_t$. Write g_t as the number of the edges between the nodes in C_t and those in the other clusters. The *structural entropy* [23] of G relative to P is

$$\mathcal{H}_P(G) = -\sum_{t=1}^{T} \frac{d_t - g_t}{2|E|} \log \frac{d_t}{2|E|}.$$

A greater value of $\mathcal{H}_P(G)$ means that the corresponding partition P gains more information about the graph G and thus P is preferable.

Given that, we would like to obtain a good partition by maximising the structural entropy $\mathcal{H}_P(G)$. Unfortunately, maximising the structural entropy $\mathcal{H}_P(G)$ is NP-hard [20, 40]. As an alternative, we propose an algorithm Clustering(G) that harnesses the power of unsupervised GCN models [40] to obtain a partition P. Specifically, Clustering(G) first employs the Singular Value Decomposition (SVD) [5] to generate spectral features, and a classical Variational Graph Auto-Encoder (VGAE) model [18] with reconstruction loss taking the generated spectral features as input to learn node embeddings. Using the obtained node embedding, Clustering(G) then trains a linear classifier to get a partition by maximising structural entropy.

Structural Informativeness. Given the learned clustering P, we propose the notion of *cluster-based marginal structural entropy* to measure the structural informativeness of data owners. Basically, the marginal structural entropy captures the information gain of a node to the clustering. The lower the marginal structural entropy of a node has, the more uncertainty this node has, and thus more information the node's data will capture. More formally, we define the marginal structural entropy of node v as the information gain due to existence of v, i.e., the difference between the structural entropy of graph G relative to P and that of G without node v relative to P'. Then the *normalised marginal structural entropy* ϵ_v of v is the normalised difference in structural entropy with partition P and another partition P' that moves v out of its cluster C_t, i.e., $\epsilon_v = (\mathcal{H}_P(G) - \mathcal{H}_{P'}(G))/\mathcal{H}_P(G)$. Let $n_{v,t}$ be the number of nodes that are incident to v and belong to C_t. After calculation, we have the following (see App. A for the detailed calculation):

Definition 4.1. The *normalised marginal structural entropy* of node $v \in C_t$ to structural entropy $\mathcal{H}_P(G)$ is

$$\epsilon_v = \frac{(d_t - g_t)\log\frac{d_t}{d_t - d_v} + 2n_{v,t}\log\frac{d_t - d_v}{2|E|}}{(d_t - g_t)\log\frac{d_t}{2|E|}}.$$

A lower normalised marginal structural entropy means more structural uncertainty of v, making v more informative.

Structural Representativeness. We use a classical structural centrality measure, PageRank [24], to quantify the structural representativeness of a node. We opt for PageRank centrality due to its superior performance compared to other centrality measures, as validated in App. D. The higher the PageRank centrality of a node has, the more representative the node is. Let $\gamma \in (0, 1)$ denote the damping factor, which controls the probability of following links. The PageRank centrality ρ_v of node v is:

$$\rho_v = \gamma \sum_{u \in N_v} \frac{\rho_u}{|N_u|} + \frac{1-\gamma}{|V|}.$$

Structural Importance score. Given the clustering P, the entropy ϵ_v and the PageRank centrality ρ_v of each node v, we define the structural importance score. Specifically, we first sort all nodes in their own cluster by their entropy and PageRank values, resp. Nodes are sorted in ascending order by entropy (as lower entropy indicates higher informativeness) and in descending order by PageRank (as higher PageRank indicates greater representativeness). This ensures that more informative and representative nodes are prioritised. For a node v in cluster C_t, let $\text{rank}_v^{\text{entr}}$ denote its rank by entropy and $\text{rank}_v^{\text{pr}}$ denote its rank by PageRank. We then define node v's informativeness and representativeness based on these rankings as

$$\phi_v^{\text{info}} := \frac{\text{rank}_v^{\text{entr}}}{|C_t|}, \text{ and } \phi_v^{\text{rep}} := \frac{\text{rank}_v^{\text{pr}}}{|C_t|}, \text{ resp.}$$

Finally, we define the structural importance score of a node. Following the approach in [7, 46], we introduce a parameter α to balance representativeness and informativeness. Intuitively, representative data helps to learn general classification patterns, while informative data is used to refine the classification boundaries. Therefore, when the budget β is relatively small compared to the overall valuations of data owners and the partition P is complex (i.e., when T is large), prioritising representative data is crucial to learning the general classification patterns. On the other hand, when the budget is relatively large and the partition is simpler, a small amount of representative data is sufficient to capture the overall pattern, allowing us to focus on acquiring more informative data to further refine the classification. More formally, given the average valuation $\overline{\theta}$, defined as the average of the upper and lower bounds of data valuations, we set $\alpha = \frac{1}{2}(1 + \frac{\beta}{n\overline{\theta}})^{-T}$. The *structural importance score* of node v is then defined as

$$\phi_v := (1-\alpha)\phi_v^{\text{rep}} + \alpha\phi_v^{\text{info}}. \quad (1)$$

4.2 Model Trading Mechanism

We propose a model trading mechanism, named *Structural importance based model trading (SIMT) mechanism*, which consists of two phases: (1) data procurement phase selects the most cost effective data owners, and (2) model training phase trains a GNN model on the procured data; See the workflow of SIMT in Figure 1 and the algorithm in Alg. 1.

Phase 1. Data Procurement. In Phase 1, SIMT takes the attributed graph $\mathcal{G}$ and the valuation vector $\boldsymbol{\theta}$, and the budget β as inputs and returns an allocation result $\boldsymbol{\pi}$ and a payment result $\boldsymbol{p}$. Firstly, Clustering(G) returns a clustering P of the nodes V after training. Given the clustering P with T clusters, the mechanism computes the structural importance score ϕ_v of each node $v \in V$ using Equation (1), which represents the importance of the node. Then an auction is conducted in each cluster $C_t \in P$ with budget β/T. In the auction for each C_t, nodes in C_t are sorted in descending order based on the ratio $\frac{\phi_v}{\theta_{i,v}}$, where data owner i owns node v. The mechanism selects the most cost-effective k data until the total payment exceeds the allocated budget. The payment to data owner i for node v is $p_{i,v} = \min\{\frac{\beta}{T}\frac{\phi_v}{\sum_{u=1}^{k}\phi_u}, \frac{\theta_{j,w}}{\phi_w}\phi_v\}$, where j is the first data owner who has not had any data selected and w is her first data in the order (if such data owner j does not exist, we set $p_{i,v}$ as $\min\{\beta\phi_v/(T\sum_{u=1}^{k}\phi_u), \tilde{\theta}\phi_v/\phi_{k+1}\}$, where $\tilde{\theta}$ is the upper bound of Θ). The total payment to data owner i is $\sum_{v \in V_i, v \le k} p_{i,v}$.

Phase 2. Model Training. Given the allocation result $\boldsymbol{\pi}$, the model training phase uses the connections, features and labels of the selected data owners to train a GNN model. However, due to budget constraints, only a subset of features and connections can be purchased, which may not be sufficient for training a robust GNN model. To address this, we first impute the missing node features

Algorithm 1 The SIMT mechanism

Input: Attributed graph $\mathcal{G}$, data owners O, valuation vector $\boldsymbol{\theta}$, and budget β
Output: Allocation $\boldsymbol{\pi}$, payment $\boldsymbol{p}$, and trained model f_{GNN}

let $G = (V, \ddot{E})$ represent the known subgraph of $\mathcal{G}$ without attributes.
Phase 1: Data procurement
get a partition $P \leftarrow \mathsf{Clustering}(G)$
compute the structural importance score ϕ_v for each $v \in V$ according to P
initialise $\boldsymbol{\pi} = \mathbf{0}, \boldsymbol{p} = \mathbf{0}$
for each cluster $C_t \in P$ **do**
  sort the nodes $v \in C_t$ by $\frac{\phi_v}{\theta_{i,v}}$ in a descending order
  find the largest k such that $\theta_k \leq \frac{\phi_k}{\sum_{u \leq k} \phi_u} \frac{\beta}{T}$
  for $v \leq k$ **do**
    Let $i \in O$ be the data owner of node v
    set $\pi_{i,v} = 1$, $p_{i,v} = \min\{\frac{\beta}{T} \frac{\phi_v}{\sum_{u=1}^{k} \phi_u}, \frac{\theta_{j,w}}{\phi_w} \phi_v\}$
procure data to get $X_s, \boldsymbol{y}_s$, update G and normalized adjacency matrix $\tilde{\boldsymbol{A}}$
Phase 2: Model training
initialise $\boldsymbol{X}' \leftarrow [X_s, \mathbf{0}_u]^\intercal$
while $\boldsymbol{X}'$ has not converged **do**
  $\boldsymbol{X}' \leftarrow \tilde{\boldsymbol{A}} \boldsymbol{X}'$
  $\boldsymbol{X}' \leftarrow [X_s, X'_u]^\intercal$
do edge augmentation on G and get $\overline{G}$
$f_{\text{GNN}} \leftarrow \mathsf{Train}(\overline{G}, \boldsymbol{X}', \boldsymbol{y}_s)$

and augment the missing edges before training. After acquiring the features from the selected nodes, we apply the feature propagation algorithm [30] to infer the features of the unselected nodes, producing a new feature matrix $\boldsymbol{X}'$. Additionally, we use the $G(n, m)$ Erdos-Renyi (ER) model [4] to generate missing edges, where m and n are determined by the edge density of the known graph. Then we incorporate contrastive learning [27] to mitigate the randomness introduced by the ER model, resulting in an augmented graph $\overline{G}$. The GNN training algorithm then takes the augmented graph $\overline{G}$, the new feature matrix $\boldsymbol{X}'$ and labels $\boldsymbol{y}_s$ as input and returns a GNN model to the consumer.

Node feature imputation. Features are crucial when training GNN [36, 38], and we apply a feature imputation method to the procured data to address missing values. Among various feature imputation methods, we choose the Feature Propagation algorithm due to its strong convergence guarantees, simplicity, speed, and scalability [30]. We use subscripts s and u to denote the selected and unselected nodes, resp. Write $\boldsymbol{X} = [\boldsymbol{X}_s, \boldsymbol{X}_u]^\intercal$ and $\boldsymbol{y} = [\boldsymbol{y}_s, \boldsymbol{y}_u]^\intercal$. Also, we write the normalised adjacency matrix $\tilde{\boldsymbol{A}}$ and the graph Laplacian Δ of $\overline{\mathcal{G}}$ as $\tilde{\boldsymbol{A}} = \begin{bmatrix} \tilde{\boldsymbol{A}}_{ss} & \tilde{\boldsymbol{A}}_{su} \\ \tilde{\boldsymbol{A}}_{us} & \tilde{\boldsymbol{A}}_{uu} \end{bmatrix}, \Delta = \begin{bmatrix} \Delta_{ss} & \Delta_{su} \\ \Delta_{us} & \Delta_{uu} \end{bmatrix}$, resp. In the feature imputation process, the feature matrix $\boldsymbol{X}'$ is initialised with the known feature $\boldsymbol{X}_s$ and a zero matrix $\mathbf{0}_u$ for the unselected nodes. The feature matrix $\boldsymbol{X}'$ is then iteratively updated as follows: $\boldsymbol{X}^{(t)} = \begin{bmatrix} \boldsymbol{I} & \mathbf{0} \\ \tilde{\boldsymbol{A}}_{us} & \tilde{\boldsymbol{A}}_{uu} \end{bmatrix} \boldsymbol{X}^{(t-1)}$, where this process continues until the feature matrix converges. The steady status of the feature matrix is [30]: $\lim_{t\to\infty} \boldsymbol{X}^{(t)} = \begin{bmatrix} \boldsymbol{X}_s \\ -\Delta_{ss}^{-1} \tilde{\boldsymbol{A}}_{us} \boldsymbol{X}_s \end{bmatrix}$.

Edge Augmentation. Given the critical role of message passing in GNNs, the absence of certain edges may impede this process, leading to sub-optimal model performance. To alleviate this issue, we introduce augmented edges to enhance message passing. Specifically, we employ the ER model [4] to generate edges for data owners with multiple unselected nodes. However, the introduction of augmented edges may inadvertently introduce noise, which could mislead the model by learning from incorrect connections. To counteract this, we integrate contrastive loss [27], denoted by L_{ctr}, into the GNN training process. This loss function encourages the model to maximise the similarity between the augmented graph and the original (non-augmented) graph views. Given a graph $\mathcal{G}$ and an augmented graph $\overline{\mathcal{G}}$, let $\boldsymbol{h}_v$ and $\boldsymbol{h}'_v$ be the feature embeddings in $\mathcal{G}$ and $\overline{\mathcal{G}}$, resp. The contrastive loss of a node v is:

$$L_{\text{ctr}}(v) = -\log \frac{\exp(\boldsymbol{h}_v \cdot \boldsymbol{h}'_v/\tau)}{\sum_{u \in V} \exp(\boldsymbol{h}_v \cdot \boldsymbol{h}'_u/\tau)},$$

where τ represents the temperature parameter, which scales the similarities between the embeddings $\boldsymbol{h}_v$ and $\boldsymbol{h}'_v$.

4.3 Analysis

Now we show that SIMT satisfies IC, IR and BF properties. Additionally, we analyse its time complexity.

THEOREM 4.2. *The SIMT mechanism is incentive compatible, individual rational and budget feasible.*

PROOF. We first show that the mechanism is IR. In each auction of cluster C_t, the utility that a data owner i obtains from a node $v > k$ when she truthfully reports is $u_{i,v}(\boldsymbol{\theta}) = \theta_{i,v}\pi_{i,v}(\theta) - p_{i,v}(\theta) = 0 - 0 \geq 0$; the utility that a data owner i obtains from $v \leq k$ is $u_{i,v}(\theta) = p_{i,v}(\theta) - \theta_{i,v}\pi_{i,v}(\theta) = \min\{(\beta\phi_v)/(T\sum_{u=1}^{k}\phi_u), (\phi_v\theta_{j,w})/\phi_w\} - \theta_{i,v} \times 1\} \geq 0$. Therefore, the utility of data owner i is $\sum_{v \in V_i} u_{i,v} \geq 0$. Also, SIMT satisfies BF. The total payment in cluster C_t is $\sum_{v=1}^{k} \min\{(\beta\phi_v)/(T\sum_{u=1}^{k}\phi_u), (\theta_{j,w}\phi_v/\phi_w\} \leq \sum_{u=1}^{k} \phi_u \times \beta/(T\sum_{u=1}^{k}\phi_u) = \beta/T$. Then the total payment in T clusters is $\leq \beta$, which shows BF.

Next we show IC. Each of data subject is assigned to an auction associated with a cluster C_t. As the assignment is independent of the reported valuation of the data owner, we just need to show that in the auction for each cluster is IC. In one auction, we consider an arbitrary data owner. When she report truthfully, there are two cases regarding each of her node, either being selected or not. We discuss the two cases separately.

(1) Consider an arbitrary node v that is selected. Assume that in the ranking, the $(k+1)$-th node is possessed by data owner j, i.e., the $(k+1)$-th ratio is $\phi_{k+1}/\theta_{j,k+1}$. Note that the data owner j could be i. If i reports a lower valuation $\theta'_{i,v} < \theta_{i,v}$ or a higher valuation $\theta_{i,v} < \theta'_{i,v} < \theta_{j,k+1}\phi_v/\phi_{k+1}$, as the marginal contribution ϕ_v is independent from the reported valuation, the ratio $\phi_v/\theta'_{i,v} \geq \phi_{k+1}/\theta_{j,k+1}$ and her ranking is still in the top k. Further, the payment of i for v is independent from i's report. As a consequence, her utility of v is $u_{i,v}(\boldsymbol{\theta}'_i, \boldsymbol{\theta}_{-i}) = u_{i,v}(\boldsymbol{\theta}_i, \boldsymbol{\theta}_{-i})$. If she reports a even higher valuation $\theta'_{i,v} \geq \theta_{j,k+1}\phi_v/\phi_{k+1}$, the ratio $\phi_{k+1}/\theta_{j,k+1} > \phi_v/\theta'_{i,v}$. Then her allocation becomes 0 and her utility of v is $u_{i,v}(\boldsymbol{\theta}'_i, \boldsymbol{\theta}_{-i}) = 0 \leq u_{i,v}(\boldsymbol{\theta}_i, \boldsymbol{\theta}_{-i})$.

(2) Consider a node v that is not selected. Assume that in the ranking, the k-th node is possessed by data owner j, i.e, the k-th ratio is $\phi_k/\theta_{j,k}$. Here, j could also be i. If i reports a higher valuation $\theta'_{i,v} > \theta_{i,v}$ or a lower valuation $\theta_{j,k}\phi_v/\phi_k \leq \theta'_{i,v} < \theta_{i,v}$, the ratio $\phi_k/\theta_{j,k} \geq \phi_v/\theta'_{i,v}$, and v's ranking is still not among the first k. Then i's utility of v is $u_{i,v}(\boldsymbol{\theta}'_i, \boldsymbol{\theta}_{-i}) = u_{i,v}(\boldsymbol{\theta}_i, \boldsymbol{\theta}_{-i}) = 0$. If i reports a much lower valuation $\theta'_{i,v} < \theta_{j,k}\phi_v/\phi_k$, and her ranking is among the first k. Her utility $u_i(\theta_i', \theta_{-i}) = \theta_i - \min\{\frac{\beta}{T}\frac{\phi_v}{\sum_{u=1}^{k}\phi_u}, \frac{\theta_{j,w}}{\phi_w}\phi_v\} \leq 0 = u_i(\theta_i, \theta_{-i})$. □

Time complexity. Given a GNN, recall m is the dimensions of the input and let m_h be the hidden layers. The computational complexity of a typical GNN is $O(|E|m + |V|mm_\mathrm{h})$. The computational complexity of SIMT is $O((|E|m+|V|mm_\mathrm{h})+(|E|m+|V|mm_\mathrm{h})+(|V|+|E|))$, where the first one terms correspond to the complexity of training the GNN, while the last two terms account for the computation of clustering and PageRank centrality resp.

5 EXPERIMENT

We conduct experiments to validate the performance of proposed SIMT mechanism in terms of node classification accuracy. (1) To demonstrate the overall performance of SIMT, we compare it with multiple baselines under different budgets. (2) To underscore the impact of each component, we perform a detailed ablation study.

5.1 Experiment Setup

Dataset. Five widely-used datasets are included in our experiments: Cora, Citeseer, Pubmed, Amazon and Coauthor [7, 17, 30, 33]. The dataset statistics are listed in Table 3 in App. B. For each dataset, we randomly sample 15% of the data as test set, which remains untouched during data procurement. This set is consistent across all baselines. Once getting the selected data from auction, we further split 80% as training data and remaining 20% for validation. To accommodate different real-world scenarios, we follow the setup in existing studies [14, 16, 22, 26, 34, 42] to validate SIMT on various datasets and varying hyperparameters.

Data valuations. We generate a set of random numbers to represent the data valuations. The valuations are sampled at random i.i.d. following a series of normal distributions $\mathcal{N}(\mu, \sigma^2)$. We get a μ drawn from $\mathcal{U}[0.8, 1.2]$ for each class to capture the difference in valuations between classes. Then for each data owner, we set the valuation of each data subject as the mean of the generated valuations of her data subjects. We set $\sigma = 0.1$. The effect of different σs on performance is investigated and the results are in App. C. To ensure all valuations are non-negative, we use a resample trick [6]. The generated valuations are in the range $[0, 2]$. Note that when the domain is different, we could scale it into $[0, 2]$.

Budget. We set the budget in $\{50, 100, 150, 200, 250, 300\}$. Given that the data valuation range is $[0, 2]$, the number of selected data is approximately from 50 to 300, which is aligned with the setup of the studies on label selection e.g. [7, 46].

Structural clustering. Here, we give the configuration of the model used for $\mathsf{Clustering}(G)$. We deploy SVD [5] to generate spectral node features, VGAE [18] to learn node embeddings followed by a linear classifier to learn the partition. We set the hidden size as 32, the learning rate as 0.01, the L2 regularisation as 5×10^{-4}. The total training budget is 400 epochs. The clustering model is initialised to solely minimise the reconstruction loss. We repeat this process 100 epochs to comprehensively capture the graph structure information. Using the obtained node embeddings, we train the linear classifier to learn a partition, maximising the structural entropy. This process is repeated 300 epochs to obtain a robust partition.

GNN model. We employ classical GNN models as f_GNN to learn node classification. Following the configurations of [17], we set the hidden size as 32, the total training budget as 200, the learning rate as 0.01 and the L2 regularisation as 5×10^{-4}. The models are optimised with minimising both reconstruction loss L_recon and classification loss L_class on the train data. We repeat 10 training iterations with different random seeds and report the average performance. To mitigate the impact of randomness in train-validation splitting, each training iteration creates 10 train-validation splits, trains 10 independent models according to the split and reports the best model according to their performance on the validation set. Ultimately, we evaluate the model performance using the test data. In other words, each experimental result is derived from 100 runs. We present the results using a GCN model and defer the exploration on the effects of different GNN architectures in App. F.

Subgraph. Each data owner possesses a subgraph with at least one node. We vary both the number o of data owners and the size n_i of theire subgraphs. We first fix the number of data owners at 10, and vary the subgraph size within $\{20, 40, 60, 80\}$ to investigate the effect of subgraph size. Next, we fix the subgraph size at 80, and vary the number of data owners within $\{5, 10, 15, 20\}$ to investigate the effect of the number of data owners. The comparison results are presented in App. E.

Baselines. To validate the overall performance of the SIMT, we benchmark it against four baseline mechanisms. These baselines incorporate different methods for assessing data importance within our proposed model trading framework. The baselines are:

- Greedy [35]: The Greedy mechanism treats all data as equally important and procures data based solely on the valuations of data owners. No feature propagation is applied.
- ASCV [8]: The ASCV mechanism first trains a VGAE model on the graph to learn node embeddings with optimising reconstruction loss, and evaluates data importance by the nodes' contribution to the reconstruction loss. The greater the contribution of the node, the more important the corresponding owner's data is. Then the auction procures data according to the ratio of data importance to valuation. No feature propagation is applied.
- Greedy(P): The Greedy(P) mechanism is the same as the Greedy except for that a feature imputation is applied.
- ASCV(P): The ASCP(P) mechanism is the same as the ASCV except for that a feature imputation is applied.

Note that ASCV is originally designed using various techniques to evaluate data importance. However, in the absence of features, only the VGAE technique can be directly applied in our scenario. For fair comparison, we redesign all baselines to avoid pre-purchase inspection of data, and set same seeds for all places involving randomness, including edge augmentation and model initialisation.

Implementation. All experiments are conducted on a single machine with AMD Ryzen 9 5900X 12-Core CPU, and NVIDIA GeForce RTX 3090 GPU. The code is implemented using Python 3.10 and Pytorch 2.0.1. Our code is available in the supplementary material.

5.2 Overall Performance

The experiment results are presented in Table 1. Here, we fix the number of data owners at 10, with each owner holding 80 data subjects. As shown in the table, SIMT consistently outperforms all

Table 1: Node classification performance under different budgets

budget		50		100		150		200		250		300		-	-
metric		MacroF1	MicroF1	MacroF1	MicroF1	MacroF1	MicroF1	MacroF1	MicroF1	MacroF1	MicroF1	MacroF1	MicroF1	average accuracy	ave. accuracy / #bought.items
Cora	Greedy	12.3 ± 4.2	21.5 ± 6.3	17.6 ± 5.6	29.1 ± 11.1	20.6 ± 6	33.9 ± 9.8	23.6 ± 6	36.7 ± 9.8	26.5 ± 5.9	40.1 ± 9.3	28.0 ± 4.7	41.5 ± 8.8	33.8	0.16
	ASCV	10.9 ± 4.5	21.0 ± 6.6	16.2 ± 5.3	29.1 ± 8.2	23.8 ± 6.9	36.0 ± 5.1	23.7 ± 4.5	36.0 ± 4.9	28.1 ± 5.6	38.5 ± 5.0	32.4 ± 8.9	43.9 ± 6.2	34.1	0.20
	Greedy(P)	24.6 ± 10.6	36.1 ± 14.4	29.0 ± 9.6	40.6 ± 12	32.3 ± 8.5	44.5 ± 10	36.0 ± 9.8	48.8 ± 8.4	37.3 ± 9.3.5	48.9 ± 8	38.6 ± 10	50.8 ± 8.3	45.0	0.22
	ASCV(P)	21.9 ± 5.8	34.8 ± 6.8	37.2 ± 11	46.9 ± 9.8	47.3 ± 7	53.7 ± 6.5	55.3 ± 5.9	61.2 ± 4.9	60.3 ± 5.1	65.2 ± 5.1	65.0 ± 6.3	68.1 ± 5.7	55.0	0.33
	SIMT	**36.1 ± 10.7**	**48.4 ± 9.5**	**46.7 ± 8.8**	**55 ± 6.7**	**56.8 ± 10**	**62.5 ± 8.6**	**62.2 ± 5.8**	**67.8 ± 3.3**	**63.4 ± 5.3**	**69.6 ± 2.8**	**66.7 ± 6.8**	**71.8 ± 4.5**	**62.5**	**0.37**
Citeseer	Greedy	6.7 ± 2.7	17.7 ± 7.2	11.0 ± 5.3	22.6 ± 8.3	14.4 ± 5.1	24.5 ± 7.8	18.7 ± 6	29.0 ± 7.8	20.5 ± 7.4	31.0 ± 8.1	21.9 ± 6.6	32.6 ± 8.3	24.6	0.12
	ASCV	8 ± 3.4	17.4 ± 7.5	12.9 ± 5.7	21.8 ± 7.4	16.9 ± 4.8	26.5 ± 4.7	20.8 ± 5.2	29.6 ± 6.6	24.3 ± 5.3	34.4 ± 5.2	28.1 ± 5.2	36.8 ± 6.4	27.8	0.17
	Greedy(P)	15.3 ± 7.5	24.5 ± 9.3	21.5 ± 8.5	30.7 ± 9.6	23.9 ± 7.9	33.2 ± 8	27.4 ± 10	36.5 ± 7.7	29.8 ± 9.8	39.0 ± 7.7	32.9 ± 11.1	42.4 ± 8.5	34.4	0.17
	ASCV(P)	19.8 ± 7.1	27.2 ± 9.5	32.2 ± 5.9	38.3 ± 9.2	38.9 ± 4	43.8 ± 6.3	42.4 ± 7	48.0 ± 7.6	46.7 ± 2.7	50.5 ± 4.5	48.6 ± 4	51.9 ± 6	43.3	0.26
	SIMT	**28.7 ± 5.8**	**39.4 ± 6.3**	**37.0 ± 5.6**	**47.4 ± 6.1**	**41.7 ± 6**	**50.2 ± 6.3**	**45.4 ± 5.2**	**53.4 ± 6.7**	**47.2 ± 5.1**	**55.8 ± 3.8**	**50.0 ± 4.9**	**57.9 ± 4.8**	**50.7**	**0.30**
Pubmed	Greedy	15.7 ± 3.8	30.8 ± 9.8	16.3 ± 3.4	30.8 ± 9.7	19.2 ± 5.9	34.0 ± 8.9	20.8 ± 7	35.0 ± 8.4	22.6 ± 9.1	37.5 ± 8.9	21.2 ± 8.3	35.8 ± 9.6	34.0	0.14
	ASCV	17.2 ± 4.3	33.7 ± 9	17.4 ± 4.7	33.9 ± 9.2	17.0 ± 4	33.8 ± 9.1	18.1 ± 5.7	34.2 ± 9.3	16.8 ± 3.8	33.8 ± 9	18.1 ± 6.2	34.5 ± 9.8	34.0	0.16
	Greedy(P)	19.7 ± 7.7	34.1 ± 9.4	25.1 ± 12.1	39.3 ± 10.5	27.6 ± 13.3	40.9 ± 11.4	29.9 ± 14.9	43.0 ± 13.3	30.6 ± 15.4	43.4 ± 13.6	31.2 ± 14.4	43.9 ± 12.6	40.8	0.17
	ASCV(P)	20.0 ± 9.8	35.8 ± 11.7	21.4 ± 12.5	36.3 ± 12.5	22.1 ± 13.5	36.9 ± 13.2	23.3 ± 15.7	37.8 ± 14.7	23.4 ± 16.1	37.7 ± 14.7	23.9 ± 16	37.9 ± 14.2	37.0	0.17
	SIMT	**22.7 ± 7.2**	**38.5 ± 7.6**	**29.9 ± 13.5**	**44.0 ± 10.6**	**36.7 ± 16.1**	**49.2 ± 11.9**	**42.0 ± 14.2**	**53.3 ± 10.5**	**43.1 ± 15.1**	**54.4 ± 11.6**	**50.2 ± 16.7**	**60.2 ± 11.1**	**49.9**	**0.25**
Amazon	Greedy	6 ± 3.2	15.5 ± 6	8.7 ± 5.4	18.2 ± 6.8	9.7 ± 4.5	17.8 ± 5.8	10.7 ± 4.8	18.5 ± 6.4	11.3 ± 5.3	19.6 ± 5.6	13.8 ± 6.1	21.3 ± 5.5	18.5	0.08
	ASCV	4.4 ± 1.5	17.3 ± 6.6	7.1 ± 3.6	20.9 ± 5.3	9.6 ± 5.2	25.2 ± 3.4	11.4 ± 7.4	27.1 ± 5.6	11.9 ± 7.2	27.3 ± 5	12.6 ± 7.2	29 ± 6	24.5	0.12
	Greedy(P)	19.6 ± 5.5	24.9 ± 6.2	23.2 ± 8	29.3 ± 8.8	23.4 ± 8.4	29.6 ± 9.1	24.8 ± 8.2	31.2 ± 8.3	24.9 ± 7.8	31.7 ± 8.9	25.4 ± 7.4	32.6 ± 8.0	30.0	0.13
	ASCV(P)	18.2 ± 7.5	31.1 ± 8.0	21.7 ± 7.3	34.8 ± 8.3	23.9 ± 8.1	36.3 ± 8.9	26.7 ± 7.0	41.9 ± 8.9	29.7 ± 9.4	43.4 ± 9.3	30.2 ± 9.1	44.7 ± 9.6	38.7	0.20
	SIMT	**30 ± 7.1**	**38.9 ± 7.9**	**38.1 ± 6.7**	**48.4 ± 6.6**	**40.7 ± 7.4**	**50.6 ± 6.7**	**44.2 ± 7.1**	**57.1 ± 5.9**	**46.2 ± 4.3**	**57.8 ± 3.6**	**51.7 ± 7.8**	**60.3 ± 6.5**	**52.2**	**0.28**
Coauthor	Greedy	8.5 ± 5.6	24.2 ± 19.2	9.2 ± 5.9	24.5 ± 19.4	11.7 ± 6.5	29.0 ± 19.8	12.9 ± 7.5	29.8 ± 20.2	15.4 ± 9.0	34.4 ± 20.2	17.1 ± 9.0	35.2 ± 20.1	29.5	0.12
	ASCV	10.3 ± 4.3	34.0 ± 17.7	11.1 ± 4.8	37.1 ± 17.7	11.8 ± 5.1	37.4 ± 17.7	12.9 ± 5.3	38.7 ± 16.7	16.1 ± 6.4	44.7 ± 14.7	16.3 ± 6.9	45.4 ± 15.1	39.5	0.18
	GreedyP	18.0 ± 8.4	34.2 ± 19.3	24.7 ± 11.3	39.6 ± 20.0	27.5 ± 12.3	42.0 ± 21.0	27.5 ± 12.3	42.1 ± 21.4	29.4 ± 13.9	44.4 ± 22.3	30.0 ± 14.1	44.1 ± 22.2	41.1	0.16
	ASCVP	16.8 ± 9.1	43.5 ± 18.6	26.2 ± 9.2	54.1 ± 13.9	25.6 ± 12.2	51.7 ± 18.9	30.7 ± 11.7	57.6 ± 15.3	30.8 ± 11.7	58.3 ± 15.4	32.1 ± 10.7	59.3 ± 15.1	54.1	0.24
	SIMT	**24.9 ± 8.9**	**55.5 ± 5.6**	**29.8 ± 9.8**	**57.9 ± 8.2**	**33.5 ± 8.4**	**61.5 ± 6.8**	**39.1 ± 7.6**	**63.4 ± 6.4**	**42.8 ± 12.0**	**65.2 ± 8.0**	**46.7 ± 11.6**	**68.8 ± 6.3**	**62.0**	**0.30**

baselines under all budgets. Compared to the vanilla Greedy and ASCV, SIMT improves up to 40% in both MacroF1 and MicroF1. Also, the last column shows the contribution per node, i.e., calculated as the average accuracy divided by the number of purchased nodes. The results consistently show that the contribution per node of SIMT is higher than that of all baselines, demonstrating the data selected by SIMT is more valuable. This validates the effectiveness of our structural importance assessment method in Sec. 4.1.

Table 1 also shows that the ASCV/ASCV(P) mechanism outperforms the Greedy/Greedy(P) mechanism. This could be attributed to that both ASCV and ASCV(P) assess the importance of data based on their structural contribution to the reconstruction loss, which, in a way, reflects structural uncertainty. However, ASCV and ASCV(P) do not perform as well as SIMT. This discrepancy underscores the effectiveness of structural importance score.

Lastly, Table 1 shows that the ASCV(P) and Greedy(P) outperform their vanilla versions by up to 20% in both MacroF1 and MicroF1. This validate the need of feature imputation.

Same trend is observed in the scenario with $n_i = 1, \forall i \in O$. See more details in Table 10 of App. F.

5.3 Ablation Study

To explore the impact of each component in SIMT on its performance, we conduct ablation studies across the five datasets and present the average test accuracy. As shown in Table 2, the four components, i.e., structuring clustering (clust), structural informativeness (info), structural representativeness (rep), and edge augmentation (edge aug), distinctly enhances SIMT's performance. In general, clust plays contributes the most among all components, which underscores the crucial role of structural clustering. Without clustering, there is a high probability that the procured data are unevenly distributed across the classes, leading to a biased training dataset. The second contributor is edge augmentation, which highlights the role of missing edge augmentation in the training

Table 2: The impact of each component

dataset	Cora		Citeseer		Pubmed		Amazon		Coauthor	
metric	acc.	Δ	acc.	Δ	acc.	Δ	acc.	Δ	acc.	Δ
no cluster	55.9	-6.6	48.0	-2.7	41.8	-8.1	35.4	-16.8	45.5	-16.5
no rep	61.0	-1.5	50.6	-0.1	48.3	-1.6	50.5	-1.7	56.9	-5.1
no info	62.3	-0.2	49.5	-1.2	47.8	-2.1	50.9	-1.3	55.1	-6.9
no edge aug	59.2	-3.3	46.2	-4.5	48.7	-1.2	41.1	-11.1	58.3	-3.7
SIMT	**62.5**		**50.7**		**49.9**		**52.2**		**62.0**	

process. Without edge augmentation, the message passing process is likely hindered, resulting in sub-optimal performance.

We also compare the effect of data valuations deviation (in App. C), graph centrality metrics (in App. D), subgraph parameters (in App. E), and GNN architectures (in App. F).

6 CONCLUSION

In this paper, we aim to design a mechanism that properly incentives data owners to contribute their data, and returns a well performing GNN model to the model consumer. In particular, we focus on the question of how we can measure data importance for model training without direct inspection. We propose SIMT, which consists of a data procurement phase and a model training phase. For data procurement, we incorporate a structure-based importance assessment method into an auction mechanism. For model training, we introduce and design two effective methods to impute missing data. As a result, SIMT ensures no data disclosure and incentive properties. Experimental results demonstrate that SIMT outperforms the baselines by up to 40% in accuracy. To the best of our knowledge, SIMT is the first model trading mechanism addressing the data disclosure problem. In the future, we will further consider the potential privacy leakage in the trained model.

ACKNOWLEDGMENTS

This work is supported by National Natural Science Foundation of China #62172077, Research Fund for International Senior Scientists #62350710215, and China Scholarship Council Grant #201906030067.

REFERENCES

[1] Jacob Abernethy, Yiling Chen, Chien-Ju Ho, and Bo Waggoner. 2015. Low-cost learning via active data procurement. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*. 619–636.

[2] Anish Agarwal, Munther Dahleh, and Tuhin Sarkar. 2019. A marketplace for data: An algorithmic solution. In *Proceedings of the 2019 ACM Conference on Economics and Computation*. 701–726.

[3] Dimitrios Bechtsis, Naoum Tsolakis, Eleftherios Iakovou, and Dimitrios Vlachos. 2022. Data-driven secure, resilient and sustainable supply chains: gaps, opportunities, and a new generalised data sharing and data monetisation framework. *International Journal of Production Research* 60, 14 (2022), 4397–4417.

[4] Béla Bollobás. 1998. *Random graphs*. Springer.

[5] Steven L Brunton and J Nathan Kutz. 2022. *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press.

[6] John Burkardt. 2014. The truncated normal distribution. *Department of Scientific Computing Website, Florida State University* 1 (2014), 35.

[7] Hongyun Cai, Vincent W Zheng, and Kevin Chen-Chuan Chang. 2017. Active learning for graph embedding. *arXiv preprint arXiv:1705.05085* (2017).

[8] Akshay L Chandra, Sai Vikas Desai, Chaitanya Devaguptapu, and Vineeth N Balasubramanian. 2021. On initial pools for deep active learning. In *NeurIPS 2020 Workshop on Pre-registration in Machine Learning*. PMLR, 14–32.

[9] Mingshu Cong, Han Yu, Xi Weng, Jiabao Qu, Yang Liu, and Siu Ming Yiu. 2020. A VCG-based Fair Incentive Mechanism for Federated Learning. (2020).

[10] Pranav Dandekar, Nadia Fawaz, and Stratis Ioannidis. 2012. Privacy auctions for recommender systems. In *Proceedings of the 8th international conference on Internet and Network Economics*. Springer-Verlag, 309–322.

[11] Amirata Ghorbani and James Zou. 2019. Data shapley: Equitable valuation of data for machine learning. In *International conference on machine learning*. PMLR, 2242–2251.

[12] Arpita Ghosh and Aaron Roth. 2011. Selling privacy at auction. In *Proceedings of the 12th ACM conference on Electronic commerce*. 199–208.

[13] Judd Randolph Heckman, Erin Laurel Boehmer, Elizabeth Hope Peters, Milad Davaloo, and Nikhil Gopinath Kurup. 2015. A pricing model for data markets. *IConference 2015 Proceedings* (2015).

[14] Rui Hu and Yanmin Gong. 2020. Trading data for learning: Incentive mechanism for on-device federated learning. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE, 1–6.

[15] Jeevan Jaisingh, Jack Barron, Shailendra Mehta, and Alok Chaturvedi. 2008. Privacy and pricing personal information. *European Journal of Operational Research* 187, 3 (2008), 857–870.

[16] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nezihe Merve Gurel, Bo Li, Ce Zhang, Costas J. Spanos, and Dawn Song. 2019. Efficient Task-Specific Data Valuation for Nearest Neighbor Algorithms. *Proceedings of the VLDB Endowment* (2019).

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

[18] Thomas N Kipf and Max Welling. 2016. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308* (2016).

[19] Avanish Kushal, Sharmadha Moorthy, and Vikash Kumar. 2012. Pricing for data markets. *online] https://courses. cs. washington. edu/courses/cse544/11wi/projects/kumar_kushal_moorthy. pdf* (2012).

[20] Angsheng Li and Yicheng Pan. 2016. Structural information and dynamical complexity of networks. *IEEE Transactions on Information Theory* 62, 6 (2016), 3290–3339.

[21] Katrina Ligett and Aaron Roth. 2012. Take it or leave it: Running a survey when privacy comes at a cost. In *International Workshop on Internet and Network Economics*. Springer, 378–391.

[22] Jinfei Liu. 2020. Dealer: end-to-end data marketplace with model-based pricing. *arXiv preprint arXiv:2003.13103* (2020).

[23] Yiwei Liu, Jiamou Liu, Zijian Zhang, Liehuang Zhu, and Angsheng Li. 2019. REM: From structural entropy to community structure deception. *Advances in Neural Information Processing Systems* 32 (2019).

[24] Nan Ma, Jiancheng Guan, and Yi Zhao. 2008. Bringing PageRank to the citation analysis. *Information Processing & Management* 44, 2 (2008), 800–810.

[25] Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a feather: Homophily in social networks. *Annual review of sociology* 27, 1 (2001), 415–444.

[26] Olga Ohrimenko, Shruti Tople, and Sebastian Tschiatschek. 2019. Collaborative machine learning markets with data-replication-robust payments. *arXiv preprint arXiv:1911.09052* (2019).

[27] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[28] Javier Parra-Arnau. 2018. Optimized, direct sale of privacy in personal data marketplaces. *Information Sciences* 424 (2018), 354–384.

[29] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. 2021. A survey of deep active learning. *ACM computing surveys (CSUR)* 54, 9 (2021), 1–40.

[30] Emanuele Rossi, Henry Kenlay, Maria I Gorinova, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M Bronstein. 2022. On the unreasonable effectiveness of feature propagation in learning on graphs with missing node features. In *Learning on Graphs Conference*. PMLR, 11–1.

[31] Aaron Roth and Grant Schoenebeck. 2012. Conducting truthful surveys, cheaply. In *Proceedings of the 13th ACM Conference on Electronic Commerce*. 826–843.

[32] Lloyd S Shapley. 1951. Notes on the n-person game—ii: The value of an n-person game. (1951).

[33] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. 2018. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868* (2018).

[34] Rachael Hwee Ling Sim, Yehong Zhang, Mun Choon Chan, and Bryan Kian Hsiang Low. 2020. Collaborative machine learning with incentive-aware model rewards. In *International conference on machine learning*. PMLR, 8927–8936.

[35] Yaron Singer. 2010. Budget feasible mechanisms. In *2010 IEEE 51st Annual Symposium on foundations of computer science*. IEEE, 765–774.

[36] Indro Spinelli, Simone Scardapane, and Aurelio Uncini. 2020. Missing data imputation with adversarially-trained graph convolutional networks. *Neural Networks* 129 (2020), 249–260.

[37] Peng Sun, Xu Chen, Guocheng Liao, and Jianwei Huang. 2022. A profit-maximizing model marketplace with differentially private federated learning. In *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 1439–1448.

[38] Hibiki Taguchi, Xin Liu, and Tsuyoshi Murata. 2021. Graph convolutional networks for graphs containing missing features. *Future Generation Computer Systems* 117 (2021), 155–168.

[39] Bisheng Tang, Xiaojun Chen, Shaopu Wang, Yuexin Xuan, and Zhendong Zhao. 2023. Generalized heterophily graph data augmentation for node classification. *Neural Networks* (2023). https://doi.org/10.1016/j.neunet.2023.09.021

[40] Yifei Wang, Yupan Wang, Zeyu Zhang, Song Yang, Kaiqi Zhao, and Jiamou Liu. 2023. User: Unsupervised structural entropy-based robust graph neural network. *arXiv preprint arXiv:2302.05889* (2023).

[41] Lei Xu, Chunxiao Jiang, Yan Chen, Yong Ren, and K. J. Ray Liu. 2015. Privacy or Utility in Data Collection? A Contract Theoretic Approach. *IEEE Journal of Selected Topics in Signal Processing* 9, 7 (2015), 1256–1269.

[42] Xinyi Xu, Lingjuan Lyu, Xingjun Ma, Chenglin Miao, Chuan Sheng Foo, and Bryan Kian Hsiang Low. 2021. Gradient driven rewards to guarantee fairness in collaborative machine learning. *Advances in Neural Information Processing Systems* 34 (2021), 16104–16117.

[43] Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. 2021. A survey on federated learning. *Knowledge-Based Systems* 216 (2021), 106775.

[44] Jingwen Zhang, Yuezhou Wu, and Rong Pan. 2021. Incentive mechanism for horizontal federated learning based on reputation and reverse auction. In *Proceedings of the Web Conference 2021*. 947–956.

[45] Mengxiao Zhang, Fernando Beltran, and Jiamou Liu. 2020. Selling Data at an Auction under Privacy Constraints. In *Conference on Uncertainty in Artificial Intelligence*. PMLR, 669–678.

[46] Wentao Zhang, Yu Shen, Yang Li, Lei Chen, Zhi Yang, and Bin Cui. 2021. Alg: Fast and accurate active learning framework for graph convolutional networks. In *Proceedings of the 2021 International Conference on Management of Data*. 2366–2374.