Pattern Recognition 153 (2024) 110567

Contents lists available at ScienceDirect

Pattern Recognition

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

Focus on informative graphs! Semi-supervised active learning for graph-level classification

Wei Ju [a], Zhengyang Mao [a], Ziyue Qiao [b], Yifang Qin [a], Siyu Yi [c], Zhiping Xiao [d], Xiao Luo [d], Yanjie Fu [e], Ming Zhang [a,*]

[a] School of Computer Science, National Key Laboratory for Multimedia Information Processing, Peking University-Anker Embodied AI Lab, Peking University, Beijing, 100871, China
[b] School of Computing and Information Technology, Great Bay University, Dongguan, 523000, Guangdong, China
[c] School of Statistics and Data Science, Nankai University, Tianjin, 300071, China
[d] Department of Computer Science, University of California, Los Angeles, 90095, USA
[e] School of Computing and Augmented Intelligence, Arizona State University, Tempe, 85280, USA

ARTICLE INFO

Keywords:
Graph classification
Graph neural networks
Semi-supervised learning
Active learning

ABSTRACT

Graph-level classification is a critical problem in social analysis and bioinformatics. Since annotated labels are typically costly, we intend to study this challenging task in semi-supervised scenarios with limited budgets. Inspired by the fact that active learning is capable of interactively querying an oracle to annotate a small number of informative examples in the unlabeled dataset, we develop a novel Semi-supervised active learning framework termed GraphSpa for graph-level classification. To make the most of labeling budgets, we propose an effective unlabeled data selection strategy that takes both local similarity and global semantic structure into account. Specifically, we first construct an adaptive queue with labeled samples and select informative samples that have a low degree of similarity to the queue using the Min-Max principle from the local view. Further, we introduce class prototypes and select samples with a large predictive loss discrepancy from the global view. To harness the full potential of unlabeled data, we develop a semi-supervised active learning framework on the basis of our fusion selection strategy coupled with graph contrastive learning during active learning. The effectiveness of our GraphSpa is validated against state-of-the-art methods through experimental results on diverse real-world benchmark datasets.

1. Introduction

A great many scenarios in the real-world are highly relevant to graph-structured data [1], such as biological networks, molecules, and traffic networks. One critical problem in modeling graph-structured data is graph-level classification, which targets at analyzing the properties of the whole graphs. This problem has a variety of downstream applications in biology and chemistry, including property prediction for molecules [2] and functionality analysis for compounds (e.g., mutagen or non-mutagen) [3].

More recently, a large number of works have been proposed [4–6] to tackle this problem. Early methods mostly leverage multiple graph kernels [7] to embed graphs into an embedding space in an unsupervised manner. Representative kernels include shortest-path kernel, random walk kernel and Weisfeiler–Lehman kernel. Unfortunately, these methods usually acquire prior knowledge from experts and thus cannot learn structural information adaptively from the data. To address this, graph neural networks (GNNs) [8,9] have been introduced into this topic to generate effective graph embeddings for downstream tasks [10–12]. Specifically, in each graph sample, a node receives information from its neighbors at each step and the neighborhood information is combined with its original representation to update the node representation. Afterward, a summarization operation is adopted to aggregate all of the updated node representations into an effective graph representation for graph-level classification.

Among the literature [4,13], GNN-based methods are usually data-hungry which indicates that they require massive labeled data to promise adequate supervisory signals. Regrettably, graph-level annotation generally requires domain experts, which are extremely expensive in specific fields [2]. For example, characterizing the properties of a simple molecule using density functional theory can often require several hours [14]. To reduce the annotation cost, two aspects may be naturally leveraged. On the one hand, it will be helpful to judiciously select the most informative unlabeled graphs for expert labeling. On the other hand, there exist a good deal of unlabeled graphs and their

* Corresponding author.
E-mail addresses: siyuyi@mail.nankai.edu.cn (S. Yi), xiaoluo@cs.ucla.edu (X. Luo), mzhang_cs@pku.edu.cn (M. Zhang).

https://doi.org/10.1016/j.patcog.2024.110567
Received 11 May 2023; Received in revised form 5 December 2023; Accepted 1 May 2024
Available online 5 May 2024

topological structures may benefit graph classification if used appropriately. Motivated by these considerations, this paper studies the problem of semi-supervised active learning for graph classification, which aims to selectively annotate the unlabeled data with limited budgets.

However, directly applying active learning techniques to graph classification is a non-trivial problem due to domain-specific characteristics. While various active learning techniques [15–17] have been developed to address this issue in other domains such as vision and natural language processing, they cannot be utilized directly for graph classification. As such, it is necessary to integrate active learning techniques into graph-level classification in a principled way due to the following reasons. To begin, learning graph-level representation is challenging since it involves dealing with a mass of graphs containing various nodes, and label scarcity would further aggravate the difficulties. Moreover, quantifying an informative graph needs to consider extra complex topological semantics of samples from both local and global perspectives. To be more specific: (i) *From a local view, an informative graph itself should be distinct from the labeled graphs*, since similar graphs tend to have analogous properties. This considers the sample–sample relationships from a local perspective; (ii) *From a global view, the representations of the whole collection of graphs should reflect the task-oriented semantic structure of the dataset*, enabling the graphs to be classified properly, and instead the graphs with fuzzy semantic prototypes are more instructive for model training. This considers the sample–class relationships from a global perspective. Towards this end, it is highly desirable to have a promising means to select informative graphs from both local and global views.

Having realized the above challenges with existing methods, in this paper, we propose a novel semi-supervised active learning framework GraphSpa for graph-level classification, which develops an effective fusion selection strategy from both local similarity and global semantic structure. On the one hand, we first employ a random walk graph kernel to calculate pairwise graph similarity and then select samples minimizing the maximum similarity between input graphs and a queue of labeled graphs following the Min-Max principle from a local view. On the other hand, we measure the semantic discrepancy of label distribution by comparing graph representations and class prototypes at different optimization steps, and then select informative graphs with larger semantic discrepancy from a global view. To fully leverage a wealth of available unlabeled graphs, we develop a semi-supervised active learning framework that augments our hybrid fusion selection strategy coupled with graph contrastive learning to further enhance the capability of semantic discrimination. In summary, we highlight our contributions as follows:

- We develop an effective fusion selection strategy to select informative unlabeled graphs for active learning, which explores graph semantics from both local and global views.
- We propose a novel semi-supervised active learning framework based on our proposed fusion selection strategy, which integrates contrastive learning and active learning for unlabeled graphs.
- Our approach has been proven superior to various state-of-the-art methods through experiments conducted across a spectrum of well-established benchmarks.

2. Related work

2.1. Graph neural networks

Graph Neural Networks (GNNs) have attracted considerable attention due to their capability of modeling graph-structured data [1,8,18]. Typically, most existing methods [5,13,19] use a neighborhood aggregation function to iteratively update the node representation by aggregating the embeddings of its neighbors, and then condensing them into a graph-level representation. For example, SUGAR [13] proposes to learn powerful representations of sampled subgraphs and incorporates self-supervised learning to enhance the performance. They obtain state-of-the-art performance due to their efficacy in learning sophisticated graph-level representations. However, these methods typically rely on supervised training, demanding extensive labeled data for optimization, a task that can be expensive and resource-intensive to annotate in real-world scenarios. With GraphSpa, apart from learning graph-level representations obtained by GNNs, we also benefit from active learning to selectively annotate informative unlabeled data with limited budgets in a semi-supervised framework.

2.2. Active learning

Our work is related to active learning, which attempts to annotate samples progressively to achieve excellent performance at a low annotation cost [15–17]. Most existing methods can be divided into three categories: uncertainty-based methods [20–22], diversity-based methods [23–25], and those that are based on model performance change [26–28]. Uncertainty-based methods select the most uncertain samples via using criteria such as maximum entropy or maximum margin. For example, Wang et al. [20] integrate uncertainty, diversity, and density in sample selection through sparse modeling using Gaussian kernels for representing the uncertainty of unlabeled data. Diversity-based methods choose diverse examples which can maximally span the input space. For instance, Wang et al. [20] introduces two diversity criteria, clustering-based and fuzzy rough set-based, for MIAL using an SVM-based MIL classifier. These criteria enhance the selection of bags by considering both informativeness and diversity. The last category assesses the future status of the model and chooses examples which enable optimal model improvement. For example, Freytag et al. [28] presents a novel active learning strategy that quantifies the expected change in model outputs, encompassing prior methods relying on expected model change and embracing the underlying data distribution. Compared with existing approaches, our GraphSpa combines the advantages of the first two methods from both global and local perspectives respectively, and focuses on tackling the challenging graph classification task with a minimal annotation cost.

2.3. Semi-supervised learning

Semi-supervised learning is another related topic to our study. Self-training has been extensively studied for a long time [29,30]. These methods mostly utilize the classifier to predict the categorization information for unlabeled samples and then utilize the well-classified samples to supervise the optimization process. For example, Noisy Student Training [30], an extension of self-training and distillation, employs larger student models and introduces noise during learning. Consistency learning is also widely used for semi-supervised learning [31,32]. These methods usually enforce the model to output consistent output after adding the perturbation. For instance, Tarvainen et al. [32] introduces Mean Teacher, a method that improves test accuracy by averaging model weights instead of label predictions and allows training with fewer labels compared to Temporal Ensembling [31]. In the literature, several works have proposed to study semi-supervised graph classification [33–36]. InfoGraph [33], GraphCL [34], JOAO [35] and DualGraph [36] both extend graph contrastive learning to semi-supervised scenarios and improve the classification performance. Additionally, there exist some algorithms combining semi-supervised learning and active learning [15–17,37,38]. For example, Gao et al. [15] propose a cost-effective approach by integrating unlabeled sample selection and model training, leveraging semi-supervised learning to distill information from both labeled and unlabeled samples. TOD [16] centers on a measure, which assesses sample loss through the evaluation of output discrepancies at various optimization steps, serving as a lower bound for accumulated sample loss. To step further, our work proposes a novel fused active selection strategy to harvest maximum gain with minimum cost while their works fail to enhance GNNs via effective interaction.

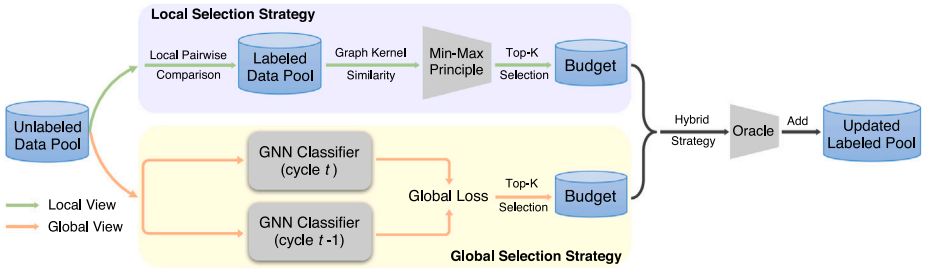

Fig. 1. An illustration of the proposed fused active selection strategy. Our fusion selection strategy actively selects informative graphs from local similarity and global semantic structure, respectively. Afterward, we combine the advantages of both worlds via a hybrid fusion strategy on unlabeled graphs and then update them to the labeled data pool.

3. Methodology

In this section, we first intuitively describe our GraphSpa and then formally present our techniques. After that, overall model optimization is introduced to perform a semi-supervised graph-level classification.

3.1. Problem formulation and preliminary

Let $G = (V, E)$ represent a graph, in which V denotes the node set and E represents the edge set. $x_v \in \mathbb{R}^F$ is adopted to represent the feature vector of v, in which F denotes the feature dimension. For active learning graph classification task, consider an unlabeled data pool of graphs $\mathcal{G}^U$, where $\mathcal{G}^U = \left\{G_1, \ldots, G_{|\mathcal{G}^U|}\right\}$, and their labels $\{y_1, \ldots, y_{|\mathcal{G}^U|}\}$ cannot be observed. In each cycle, we aim to select a fixed budget of examples from the unlabeled data pool $\mathcal{G}^U$ and the chosen examples will be annotated using an oracle, and then added into the labeled data pool $\mathcal{G}^L$. The budget size $\mathcal{B}$ is fixed, which is generally much smaller than the size of the unlabeled data pool $\mathcal{G}^U$. Our purpose is to learn a graph-level prediction model $\mathcal{H} : G \to y$ by selecting the most informative unlabeled examples for additional annotation.

3.2. Overview

This paper proposes a semi-supervised active learning framework GraphSpa as shown in Fig. 1. Previous methods utilize semi-supervised learning to overcome the scarcity of data annotations in the graph domain, which could suffer from biased and overconfident pseudo-labels [2]. To tackle this, we propose to select informative graphs via active learning, which would facilitate both industry and academic applications in practice. The core of GraphSpa is to study local similarity and global semantic structure of the graphs, such that we can actively select informative graphs. Specifically, GraphSpa explores local similarity by a non-parameterized random walk kernel while the global semantic structure is modeled via multiple prototypes. Further, we integrate our fusion selection strategy into a unified semi-supervised framework. Supervised learning and contrastive learning are combined jointly to enhance the model optimization and improve the performance.

3.3. GNN encoder

Recently, graph neural networks [8] have gained increasing popularity owing to their powerful ability to learn structured data, which is capable of embedding graph structure into the learned node representations via message passing mechanism [39]. Specifically, the representation vector of a node v at layer k is represented by $\mathbf{h}_v^{(k)}$. For each node $v \in V$, first the representations from its neighbors at layer $k-1$ would be aggregated. Then, the representation $\mathbf{h}_v^{(k)}$ would be updated by combining the node representation of v in the previous layer with the aggregated neighbor representation in an iterative manner. Formally, $\mathbf{h}_v^{(k)}$ is calculated as:

$$\mathbf{h}_v^{(k)} = C_\theta^{(k)}\left(\mathbf{h}_v^{(k-1)}, \mathcal{A}_\theta^{(k)}\left(\left\{\mathbf{h}_u^{(k-1)}\right\}_{u\in\mathcal{N}(v)}\right)\right), \tag{1}$$

in which $\mathcal{N}(v)$ collects the neighbors of v. Here $\mathcal{A}_\theta^{(k)}$ and $C_\theta^{(k)}$ denotes the aggregation and combination functions at the kth layer, respectively. At last, we derive the graph-level embedding vector by aggregating all node embeddings at the last layer using a readout operation as follows:

$$f_\theta(G) = \text{READOUT}\left(\left\{\mathbf{h}_v^{(K)}\right\}_{v\in V}\right), \tag{2}$$

in which $f_\theta(G)$ is the graph-level representation, θ is the parameter of encoder. The readout operation is taking the sum of all the node representations in our implementation following recent works [40].

3.4. Local selection strategy

Intuitively, graphs with similar local structures should have similar properties. On the contrary, a graph rich in information should be dissimilar to existing labeled graphs to seek the maximum gain for performance. For this purpose, we attempt to select informative graphs based on local similarity. Specifically, we first construct an adaptive queue $\mathcal{G}_q$ that is randomly selected from the labeled data as anchor graphs, and then updated following a first-in, first-out principle. Inspired by the Min-Max principle [41], we select the unlabeled graphs minimizing the maximum similarity between the input graphs and the queue of labeled graphs, such that the selected graphs can better represent the data distribution, providing more richness and diversity with the same budget. Formally, we define a measurement indicating the similarity from each input G to the queue:

$$\phi_l(G) = \max_{G'\in\mathcal{G}_q} \text{Sim}(G, G'), \tag{3}$$

where $\text{Sim}(\cdot,\cdot)$ measures the similarity of two graphs and $G' \in \mathcal{G}_q$ is the graph in the queue. The graphs in the unlabeled pool with lower local similarity will be selected for annotation and added to the labeled pool. In other words, we choose a subset $\mathcal{B}_l$ in the unlabeled pool such that $\sum_{G\in\mathcal{B}_l}\phi_l(G)$ is minimized according to the Min-Max principle [41]. Specifically, we arrange the graph samples from the unlabeled data pool into an ordered set using the aforementioned local selection strategy. For different datasets, we apply appropriate thresholds to select a subset that meets the criteria, forming the final $\mathcal{B}_l$. As limited annotation information is available, we measure the pairwise similarity in a non-parameterized manner. Moreover, to capture the local structure more effectively, we propose to employ the random walk kernel, which is widely used in graph matching [7], to explore graph topology information for similarity measurement.

Specifically, we first review the definition of graph direct product. Considering two graph samples $G = (V, E)$ and $G' = (V', E')$, their direct product $G_\times = (V_\times, E_\times)$ is still a graph in which $V_\times = \{(v, v') : v \in V \wedge v' \in V'\}$ and $E_\times = \{\{(v, v'), (u, u')\} : \{v, u\} \in E \wedge \{v', u'\} \in E'\}$. It has been proven that conducting a random walk on direct product $G_\times$ of G and G' is equivalent to running a concurrent random walk on two original graphs [42]. Note that traditional random walk kernels can count all pairs of matching walks on G and G'. On this basis, the number of matching random walks could be derived from

the adjacency matrix $\mathbf{A}_\times$ when the starting and stopping probabilities over nodes in original graphs are from uniform distributions. Then, the P-step random walk kernel between G and G' can be written as:

$$k\left(G, G'\right) = \sum_{i=1}^{|V_\times|} \sum_{j=1}^{|V_\times|} \left[\sum_{p=0}^{P} \lambda_p \mathbf{A}_\times^p\right]_{ij}, \tag{4}$$

in which $\lambda_0, \ldots, \lambda_P$ are positive, real-valued weights. Eq. (4) indicates that the random walk kernel $k\left(G, G'\right)$ considers the sum of kernel values for the number of common walks of length from 1 to P. However, considering all lengths simultaneously can incur a certain computational cost. To improve computational efficiency, we exactly calculate the number of common walks with length p between two graphs, where we set $\lambda_P = 1$ in this case:

$$\mathrm{Sim}(G, G') = k^{(p)}\left(G, G'\right) = \sum_{i=1}^{|V_\times|} \sum_{j=1}^{|V_\times|} \left[\mathbf{A}_\times^p\right]_{ij}. \tag{5}$$

In this manner, we are capable of efficiently selecting informative graphs by capturing topology information with a non-parameterized random walk kernel from the perspective of the local structure similarity. For example, structural information indicates the property of molecules and proteins, which is very crucial for effective graph classification. Our local selection strategy aims to select informative graphs away from labeled graphs from the topological view.

3.5. Global selection strategy

For the entire set of graphs, effective graph representations in the embedding space should be able to reflect the global semantic structure of the data, so that the graph samples with similar semantic properties such as the same class label are compactly embedded. To this end, we introduce an additional set of model parameters to represent the class prototypes of different labels in the latent space. They are formally defined as $C = \{\mathbf{c}_l\}_{l=1}^{L}$ where L denotes the class number. The goal of global-semantic learning is to encourage the graphs to be embedded close to each other around corresponding class prototypes. After obtaining embedding $\mathbf{z}$ of each graph G based on the GNN-based encoder, the assignment probability for each graph is formalized as:

$$P(y = l|G) = \frac{\exp(\mathbf{z}^\top \mathbf{c}_l)}{\sum_{l'=1}^{L} \exp(\mathbf{z}^\top \mathbf{c}'_l)}. \tag{6}$$

where the prototypes $\{\mathbf{c}_l\}_{l=1}^{L}$ in our global selection strategy is used to represent the centroids of different classes in the latent space. In our implementation, we randomly initialize these prototypes and then update them using Adam optimizer by minimizing the cross-entropy loss using Eq. (6) on labeled graphs.

Following [15], we believe that labeling examples with highly inconsistency should of great value since these examples are different to be optimized without supervised loss. Instead, querying an oracle to annotate these challenging samples can ensure the correctness of labels, and enable them to be beneficial for model training at the next cycle. Motivated by this, we introduce a simple measurement to calculate the inconsistency of global predictions:

$$\phi_g(G) = \|p(y|G; \boldsymbol{\Phi}_t) - p(y|G; \boldsymbol{\Phi}_{t-1})\|, \tag{7}$$

where $\boldsymbol{\Phi} = \{\theta, C\}$ denotes the set of the whole parameters and $\boldsymbol{\Phi}_t$ implies the parameters at the end of tth cycle. $p(y|G; \boldsymbol{\Phi}_t) = [p(y = 1|G; \boldsymbol{\Phi}_t), \ldots, p(y = L|G; \boldsymbol{\Phi}_t)]$ is the label distribution. The measurement calculates the difference between assignment probabilities between two cycles, which implies the stability of the assignment probability for each graph. For this reason, the graphs in the unlabeled pool with higher inconsistency will be regarded as rich in information. In other words, we aim to choose a subset $\mathcal{B}_g$ such that $\sum_{G \in \mathcal{B}_g} \phi_g(G)$ is maximized. Specifically, we arrange the graph samples from the unlabeled data pool into an ordered set using the aforementioned global selection strategy. For different datasets, we apply appropriate thresholds to select a subset that meets the criteria, forming the final $\mathcal{B}_g$. In this way, we are capable of selecting informative graphs by exploring the graph's semantic properties via multiple class prototypes from a global perspective. Our global selection strategy is close to graph classification since it focus on the assignment probability of each graph directly. Take an example, when we cannot get consistent knowledge for some molecules during our learning, we would consider them informative.

3.6. Hybrid fusion strategy

Either of the two selection strategies proposed above has its inherent interest preference. Intuitively, we expect to sample unlabeled graphs under the collaboration between both selection strategies to overcome instability and bias. In particular, we develop three different hybrid fusion strategies to combine the advantages of both worlds and couple the information from both local and global perspectives for effective active learning.

Intersection. A straightforward fusion way is to select the informative graph only if it is considered reliable by both strategies. Specifically, we first choose subset $\mathcal{B}_l$ and $\mathcal{B}_g$ following ϕ_l and ϕ_g according to Sections 3.4 and 3.5. Here, taking $\mathcal{B}_l$ as an example, we arrange the graph samples from the unlabeled data pool into an ordered set using the aforementioned local selection strategy. For different datasets, we apply appropriate thresholds to select a subset that meets the criteria, forming the final $\mathcal{B}_l$. On the other hand, $\mathcal{B}_g$ is obtained using the aforementioned global selection strategy, and then select the subset $\mathcal{B}_h = \mathcal{B}_l \cap \mathcal{B}_g$ as the final informative graphs to be annotated by the oracle, here the size of $\mathcal{B}_h$ is equal to the number of graph samples required in each cycle of the active learning strategy. Note that when two sets have no interaction, we will attempt to loose the condition to enlarge both two sets.

Union. It seems that directly using the intersection set of global and local graph sets may ignore some good informative samples, leading to some information loss and suboptimal performance. To this end, an alternative fusion way is to select the informative graphs by both strategies and choose the subset $\mathcal{B}_h = \mathcal{B}_l \cup \mathcal{B}_g$ as the final informative graphs, where the acquisition of $\mathcal{B}_l$ and $\mathcal{B}_g$ is obtained is same as the way in **Intersection**, except for selecting appropriate threshold conditions, such that the size of $\mathcal{B}_h$ is equal to the number of graph samples required in each cycle of the active learning strategy.

Attention. It is often not appropriate to select the informative graphs through the fixed strategies, so we further introduce some weights on the two strategies as additional model parameters which are updated during model training. Formally, we introduce a learnable weight vector $\mathbf{w} = [w_1, w_2]$. As the weights are tailored for two different strategies and shared across different graphs, this hybrid fusion strategy ensures both flexibility and effectiveness. Specifically, our attention strategy outputs the final score:

$$\phi_a(G) = \sigma(w_1 \phi_l(G) + w_2 \phi_g(G)), \tag{8}$$

where σ denotes the sigmoid function. The ground truth of $\phi_a(G)$ is defined as the $1 - P(y = y_G|G)$ where y_G is the label of G since harder samples with lower predictive accuracy indicate high values for active learning. In practice, we make use of validation data with regression loss for optimization of the weight vector and then similarly choose a subset $\mathcal{B}_h$ which maximizes $\sum_{G \in \mathcal{B}_h} \phi_a(G)$, where the size of $\mathcal{B}_h$ is equal to the number of graph samples required in each cycle of the active learning strategy.

Based on these three different fusion strategies, the selected graphs can enhance the performance more effectively and efficiently. Besides, note that our hybrid fusion strategy combines the diversity-based method (i.e., the local strategy) and the uncertainty-based method (i.e., the global strategy), which can provide a comprehensive criterion to select more valuable informative graphs to benefit the process of model training.

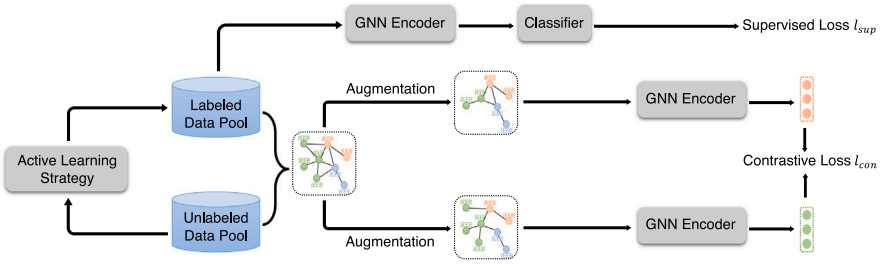

Fig. 2. An illustration of a semi-supervised active learning framework. The framework coupled with our fusion selection strategy in the active learning module is optimized by supervised loss as well as contrastive loss based on graph representations in the embedding space via graph augmentations.

3.7. Semi-supervised active learning framework

In this section, we integrate the above hybrid fusion strategy into a unified semi-supervised active learning framework in an effective way, as shown in Fig. 2. There are always a wealth of available unlabeled graphs that usually exist in many domains. Although their labels are unknown, the structures of these unlabeled graphs can usually be adopted to help learn effective graph-level representations. Towards this end, we seek to leverage this information to further overcome the severe label scarcity.

Contrastive Learning. Inspired by recent graph contrastive learning [34,35], we attempt to fully leverage the unlabeled data with contrastive learning to enhance the model training. Specifically, our model involves graph augmentations to build generalized graph-level representation pairs. Typically, there are four basic data augmentation strategies: (1) *Edge deletion* eliminates several edges from the graph at random. (2) *Node deletion* samples several nodes and eliminates them and their connected edges from the graph. (3) *Attribute masking* masks partial attributes of selected vertices at random. (4) *Subgraph* selects a subgraph using random walk. In practice, we build augmented graphs by choosing one of these operations at random.

To begin, we conduct the above stochastic graph augmentations for each graph, producing a positive pair, i.e., $\hat{G}_i$ and $\hat{G}_j$. After then, the GNN-based encoder $f_\theta(\cdot)$ is used to extract graph-level representations $\mathbf{z}_i$ and $\mathbf{z}_j$ for augmented graphs $\hat{G}_i$ and $\hat{G}_j$. We adopt the noise-contrastive estimation loss [43], which encourages us to enlarge the similarity between positive pairs, i.e., $\{\mathbf{z}_i, \mathbf{z}_j\}$ with the comparison to negative pairs. To generate negative pairs, we first construct a mini-batch containing M graphs, which results in $2M$ augmented samples, i.e., $\{\hat{G}_{m,i}, \hat{G}_{m,j}\}_{m=1}^{M}$. Then for each positive pair $\hat{G}_{m,i}$ and $\hat{G}_{m,j}$, the other $M-1$ augmented graphs in the minibatch are considered as negatives. Let $\mathbf{z}_{m,i} \star \mathbf{z}_{m,j}$ compute the similarity between $\mathbf{z}_{m,i}$ and $\mathbf{z}_{m,j}$. If $\mathbf{z}_i$ and $\mathbf{z}_j$ are re-annotated as $\mathbf{z}_{m,i}$ and $\mathbf{z}_{m,j}$ for the mth graph, respectively, we compare two graph representations for the mth graph:

$$\ell_{con} = -\log \frac{e^{\mathbf{z}_{m,i} \star \mathbf{z}_{m,j}/\tau}}{\sum_{m'=1}^{M} e^{\mathbf{z}_{m,i} \star \mathbf{z}_{m',j}/\tau}}, \tag{9}$$

where τ is a temperature parameter set to 0.5 following [34,35].

Supervised Learning. At each cycle, we minimize the supervised objective given the labeled set $\mathcal{G}^L$ as follows:

$$\ell_{sup} = \frac{1}{|\mathcal{G}^L|} \sum_{G_j \in \mathcal{G}^L} \left[-\log p\left(y_j \mid G_j\right) \right], \tag{10}$$

where in the first cycle, the model is trained by annotating a random subset of the unlabeled data.

Joint Optimization Loss. By integrating the supervised learning loss and self-supervised contrastive learning loss, we minimize an overall learning objective at each cycle as follows:

$$\ell = \ell_{sup} + \ell_{con}. \tag{11}$$

Algorithm 1 Learning Algorithm of GraphSpa

Input: Unlabeled pool $\mathcal{G}^U$. The total number of cycles R.
Parameter: GNN module parameter θ. Class prototype parameter C.
Output: Jointly learned $p(y|G)$

```
Initialize model parameter.
Sample samples from G^U and add to labeled pool G^L.
for r = 1, 2, ..., R do
    while not convergence do
        Forward propagation through graph augmentation and GNN-based encoder.
        /* Eq. (11)                                                    */
        Calculate loss function in Eq. (11).
        Update parameters through back-propagation.
    end while
    /* Eq. (3), Eq. (7)                                                */
    Choose subset B_l and B_g following φ_l and φ_g. /* Eq. (8)        */
    Select subset B_h through hybrid fusion strategy.
    Update queue with B_h following a first-in, first-out manner.
end for
```

Table 1
Statistics of the datasets.

Datasets	Graph Num.	Avg. nodes	Avg. edges	Classes
PROTEINS	1113	39.06	72.82	2
DD	1178	284.32	715.66	2
IMDB-B	1000	19.77	96.53	2
IMDB-M	1500	13.00	65.94	3
REDDIT-B	2000	429.63	497.75	2
REDDIT-M-5k	4999	508.52	594.87	5

In this way, we are able to fully leverage the unlabeled data combined with our effective fused active selection strategy in our semi-supervised active learning framework. The training procedure of our GraphSpa is shown in Algorithm 1.

4. Experiment

4.1. Experimental settings

Datasets. To evaluate the effectiveness of our GraphSpa, we conduct experiments on six benchmark datasets [44], including two bioinformatics datasets (i.e., PROTEINS and DD), four social network datasets (i.e., IMDB-B, IMDB-M, REDDIT-B, and REDDIT-M-5k). The statistics of these datasets are summarized in Table 1. Following previous works [33], we use all-ones embeddings as initial node features if their attributes are not accessible. For each dataset, we randomly select 70% and 20% of the whole data to constitute the train set and test set, and treat the remaining as validation set to tune hyper-parameters. We allocate 1/7 of the train set (i.e., 10% of the whole dataset) as a budget

Table 2
Performance comparison on six benchmark datasets over five runs (in %).

Method	PROTEINS	DD	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M-5k
GK [45]	64.8 ± 2.3	53.2 ± 1.4	54.5 ± 1.7	32.3 ± 2.4	57.8 ± 2.7	34.3 ± 0.8
SP [46]	65.2 ± 2.6	55.3 ± 2.1	52.0 ± 1.6	37.7 ± 1.9	68.3 ± 3.7	30.4 ± 1.3
WL [47]	63.5 ± 1.6	57.3 ± 1.2	58.1 ± 2.3	33.3 ± 1.4	61.8 ± 1.3	37.0 ± 0.9
DGK [48]	64.4 ± 1.7	60.5 ± 0.8	55.6 ± 2.2	34.6 ± 1.3	66.2 ± 2.4	36.5 ± 2.4
Sub2Vec [49]	52.7 ± 4.5	46.4 ± 3.2	44.9 ± 3.5	31.8 ± 2.7	63.5 ± 2.3	35.1 ± 1.5
Graph2Vec [50]	63.1 ± 1.8	53.7 ± 1.6	61.2 ± 2.6	38.1 ± 2.2	67.7 ± 2.3	38.1 ± 1.4
EntMin [29]	62.7 ± 2.7	59.8 ± 1.3	67.1 ± 3.7	37.4 ± 1.2	66.9 ± 3.5	38.7 ± 2.8
Π-Model [32]	63.2 ± 1.2	61.8 ± 1.8	67.0 ± 3.4	39.0 ± 3.5	67.1 ± 2.9	39.0 ± 1.1
Mean-Teacher [32]	64.3 ± 2.1	60.6 ± 1.8	66.4 ± 2.7	38.8 ± 3.6	68.7 ± 1.3	39.2 ± 2.1
VAT [51]	64.1 ± 1.2	59.9 ± 2.6	67.2 ± 2.9	39.6 ± 1.4	70.8 ± 4.1	38.9 ± 3.2
InfoGraph [33]	68.2 ± 0.7	67.5 ± 1.4	71.8 ± 2.3	42.3 ± 1.8	75.2 ± 2.4	41.5 ± 1.7
GraphCL [34]	69.4 ± 0.8	68.7 ± 1.2	71.2 ± 2.5	43.7 ± 1.3	75.2 ± 1.7	42.3 ± 0.9
JOAO [35]	68.7 ± 0.9	67.9 ± 1.3	71.0 ± 1.9	42.6 ± 1.5	74.8 ± 1.6	42.1 ± 1.2
DualGraph [36]	70.1 ± 1.2	69.8 ± 0.8	72.1 ± 0.7	**44.8 ± 0.4**	75.4 ± 1.4	42.9 ± 1.4
GHNN [52]	71.1 ± 0.3	70.6 ± 0.4	72.3 ± 0.6	42.8 ± 0.4	76.3 ± 0.7	**44.1 ± 0.5**
ASGN [2]	67.7 ± 1.2	68.5 ± 0.6	70.6 ± 1.4	41.2 ± 1.4	73.1 ± 2.3	42.2 ± 0.8
MCDAL [22]	70.7 ± 1.0	69.8 ± 0.8	72.0 ± 1.3	42.3 ± 0.9	75.2 ± 0.9	42.9 ± 0.8
GALAXY [53]	70.2 ± 0.5	70.3 ± 0.7	70.8 ± 0.8	43.5 ± 1.3	75.3 ± 0.6	43.4 ± 0.4
ASGNN [54]	71.0 ± 0.7	71.1 ± 0.9	71.0 ± 1.0	44.1 ± 0.7	73.5 ± 0.7	43.2 ± 0.5
GraphSpa	**71.2 ± 0.7**	**71.4 ± 0.8**	**72.3 ± 1.1**	44.5 ± 0.6	**76.5 ± 0.4**	44.0 ± 0.6
p-value	0.08	0.03	0.18	0.42	0.04	0.21

available for label annotation, while the remaining data in the train set is considered as the unlabeled set.

Baselines. To show the superiority of our approach, we compare our GraphSpa with competitive baselines which can be boiled down to four categories, i.e., traditional graph approaches, traditional semi-supervised approaches, graph-specific semi-supervised approaches and active learning approaches. Traditional graph approaches include Graphlet Kernel (GK) [45], Shortest Path Kernel (SP) [46], Weisfeiler–Lehman (WL) Kernel [47], DGK [48], Sub2Vec [49], and Graph2Vec [50]. Traditional semi-supervised approaches include EntMin [29], Π-Model [32], Mean-Teacher [32] and VAT [51]. Graph-specific semi-supervised approaches include InfoGraph [33], GraphCL [34], JOAO [35], DualGraph [36], and GHNN [52]. Active learning approaches include ASGN [2], MCDAL [22], GALAXY [53] and ASGNN [54].

Parameter Settings. All the experiments are implemented using PyTorch. Following previous works [33], GIN [40] is adopted as the GNN encoder f_θ. We search for the optimal parameters on the validation set and evaluate the model on the test set. The total number of active learning cycles R is set to 9, while the number of data samples queried in each cycle is set to $\mathcal{B}/(R+1)$. The random walk length p is set to 3. To promise a fair comparison, the batch size is set to 64 and the total number of epochs is set to 100 for all datasets. The dimension of hidden embeddings is set to 64 for all datasets. We use the "Intersection" as our default hybrid strategy in our experiment. The parameters for all baseline approaches are carefully tuned following their corresponding papers to achieve optimal performance.

4.2. Performance comparison

In Table 2, we summarize the quantitative findings of semi-supervised graph classification. Here we compare our method GraphSpa with all baseline methods in a fair setting. For example, when we have 10% of the total dataset as labeling budgets, GraphSpa, ASGN, MCDAL, GALAXY and ASGNN start the training with a randomly selected 1% labeling budgets and conduct active learning selection until 10% labeling budgets are utilized, while InfoGraph, GraphCL, JOAO, DualGraph and GHNN are directly trained with 10% labeling budgets. Note that the training of the latter does not involve active learning, but all methods are conducted with the same budgets for a fair comparison. From the comprehensive views, we have the following observations:

- The majority of traditional graph methods are inferior to other approaches, which indicates that these graph methods may be ineffective in capturing effective information via GNNs. Moreover, features in these methods are typically heuristic, which results in worse generalization ability.
- A general observation is that graph-specific semi-supervised learning approaches perform better than traditional semi-supervised learning techniques by a significant margin, which verifies that models specifically designed for graph-structured data have strong representation capability in capturing effective information of the graph topology and node attributes.
- By incorporating contrastive learning into GNNs, the recent state-of-the-art method GHNN has obtained high enough performance, which pushes away the other graph-specific semi-supervised learning baselines (InfoGraph, GraphCL, JOAO and DualGraph), sufficiently showing the effectiveness of the instance discrimination principle for contrastive learning and complementary two-branch learning framework.
- For four active learning baselines, the latest ASGNN achieves the best results on most datasets. Similar to our framework, it also simultaneously considers the uncertainty of sample predictions and selects representative samples with diversity, maximizing the effectiveness of the active learning strategy. The other three baselines (ASGN, MCDAL and GALAXY) only take into account partial factors, leading to sub-optimal results.
- Overall, from the results, it can be observed that our framework GraphSpa outperforms the baselines on most datasets, showing the superiority and efficacy of our approach. We attribute the performance gain to two factors: (i) The effective sample selection strategy. Our selection strategy explores both local similarity and global semantic structure, sampling informative graphs for annotation. (ii) The semi-supervised active learning framework. We integrate both self-supervised learning and active learning in a principled manner which can be beneficial for classification.
- We have conducted statistical analysis of Wilcoxon tests to justify that the gains with the best baseline are statistically significant with p-value < 0.1. From the Table 2, We observed statistically significant improvements in the performance of our model on three out of six datasets. The lack of significance on the remaining datasets may be attributed to the limited gains achieved solely through active learning, as our base model is relatively basic. Introducing more sophisticated self-supervised techniques to fully harness unlabeled graphs might further enhance performance.

Table 3
Performance comparison under different dataset split settings over five runs (in %).

Methods	PROTEINS	DD	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M-5k
MCDAL	70.4 ± 1.1	70.8 ± 2.1	70.0 ± 2.7	44.1 ± 2.1	75.6 ± 1.0	43.2 ± 1.4
ASGNN	71.2 ± 1.6	71.0 ± 1.6	69.1 ± 3.0	43.9 ± 1.4	75.3 ± 0.9	43.6 ± 1.0
GraphSpa	**71.8 ± 0.9**	**71.9 ± 1.2**	**70.6 ± 2.7**	**44.6 ± 1.1**	**76.1 ± 0.9**	**44.2 ± 0.8**

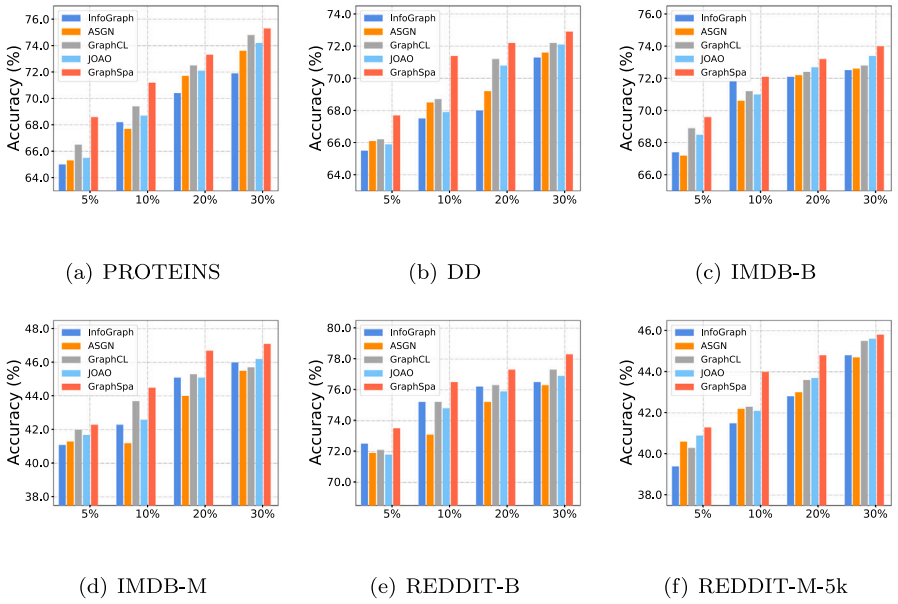

Fig. 3. Performance on datasets *w.r.t.* the amounts of annotation budget (i.e., 5%, 10%, 20%, 30%) and all the unlabeled data.

- Finally, we analyze the impact of different data splits in Table 3. We conduct five random splits, recording the mean and standard deviation. Here, we compare our GraphSpa model with the two latest methods (MCDAL and ASGNN), and the results consistently demonstrate our model's superiority across all datasets. This further showcases the robustness and excellence of our framework and the proposed active learning strategy.

Influence of different labeling budget rates. In this section, we show the model performance with different rates of labeling budgets (i.e., labeled data). As illustrated in Figure Fig. 3, the following observations can be inferred from the results:

- Overall, the findings indicate that the performance of all methods improves with the increase of the number of accessible labeling budgets. The reason is that graph classification methods are inherently data-driven, and labeling budgets contain the most discriminative signals for category analysis, showing that increasing the number of labeling budgets is an effective way for training.
- Among all the methods, our GraphSpa consistently achieves the best results with the increase of labeling budgets, which indicates that actively selecting informative graphs via our proposed strategy further improves ability by selecting the most representative samples with minimal labeling costs, thereby outperforming the baselines with an even greater margin.

Effectiveness analysis of the proposed active learning methods. To better illustrate the effectiveness and superiority of the active learning strategy proposed in our framework, we combine some representative baseline methods (InfoGraph and GraphCL) with our proposed active learning strategy. In other words, we train these baseline methods along with the graph samples selected through active learning in our GraphSpa for fair comparison. As shown in Table 4, we observe consistent performance improvements when both GraphCL and InfoGraph are equipped with our active learning strategy, emphasizing the effectiveness of our proposed active learning strategy. However, we find that GraphCL with A outperforms our GraphSpa on certain datasets, which is natural as our framework employs a basic model combined with an active learning strategy for training, while other baselines incorporate their respective more complex techniques. Nevertheless, our method still achieves optimality on many datasets, further affirming the superiority of our active learning strategy.

Table 4
Effectiveness analysis of the proposed active learning module (in %).

Methods	PROTEINS	DD	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M-5k
InfoGraph	68.2 ± 0.7	67.5 ± 1.4	71.8 ± 2.3	42.3 ± 1.8	75.2 ± 2.4	41.5 ± 1.7
InfoGraph w A	70.3 ± 0.8	69.1 ± 1.5	72.0 ± 1.8	43.2 ± 1.6	76.3 ± 1.9	42.9 ± 1.5
GraphCL	69.4 ± 0.8	68.7 ± 1.2	71.2 ± 2.5	43.7 ± 1.3	75.2 ± 1.7	42.3 ± 0.9
GraphCL w A	**71.7 ± 1.0**	71.0 ± 1.1	72.2 ± 1.9	44.0 ± 1.0	**77.1 ± 1.7**	**44.1 ± 0.9**
GraphSpa	71.2 ± 0.7	**71.4 ± 0.8**	**72.3 ± 1.1**	**44.5 ± 0.6**	76.5 ± 0.4	44.0 ± 0.6

4.3. Ablation study

In this section, we investigate a few variants to demonstrate how every part of our model affects the performance:

- **GNN-Sup.** Our base model, which trains a GNN solely on initial random labeled data in a fully supervised way.
- **GraphSpa w/o A.** We do not use an active learning strategy to select data for annotation.
- **GraphSpa w/o L.** We remove the local selection strategy and only adopt the global selection strategy.
- **GraphSpa w/o G.** We remove the global selection strategy and only adopt the local selection strategy.
- **GraphSpa w/o C.** We remove the contrastive learning loss and only adopt a hybrid active learning strategy.

Table 5
Ablation study of several model variants (in %).

Methods	PROTEINS	DD	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M-5k
GNN-Sup	63.3 ± 1.4	62.5 ± 1.5	63.4 ± 2.1	39.2 ± 1.6	69.8 ± 1.1	38.6 ± 2.5
GraphSpa w/o A	66.7 ± 1.6	65.4 ± 1.7	64.5 ± 1.3	41.2 ± 1.1	71.3 ± 0.8	36.7 ± 1.3
GraphSpa w/o L	70.1 ± 1.3	69.8 ± 1.2	70.2 ± 1.0	40.6 ± 1.7	74.5 ± 1.4	39.9 ± 1.0
GraphSpa w/o G	69.6 ± 1.2	69.3 ± 1.1	71.1 ± 0.7	42.0 ± 1.8	72.3 ± 1.6	36.4 ± 1.8
GraphSpa w/o C	70.7 ± 1.3	70.2 ± 1.2	71.7 ± 1.0	43.5 ± 2.8	75.8 ± 2.2	42.1 ± 1.5
Full model (Ours)	**71.2 ± 0.7**	**71.4 ± 0.8**	**72.3 ± 1.1**	**44.5 ± 0.6**	**76.5 ± 0.4**	**44.0 ± 0.6**

Table 6
Performance *w.r.t.* the embedding dimensions on all datasets (in %).

Embedding dimensions	8	16	32	64	128	256
PROTEINS	66.8 ± 1.7	68.8 ± 0.9	70.3 ± 1.4	71.2 ± 0.7	71.0 ± 1.2	70.2 ± 1.3
DD	70.6 ± 1.2	71.3 ± 0.8	70.8 ± 1.1	71.4 ± 0.8	71.1 ± 0.7	71.7 ± 0.9
IMDB-B	68.2 ± 1.4	70.3 ± 1.2	72.2 ± 0.9	72.3 ± 1.1	72.5 ± 1.0	72.3 ± 0.8
IMDB-M	39.3 ± 0.6	41.4 ± 0.9	43.8 ± 0.5	44.5 ± 0.6	44.1 ± 0.8	44.3 ± 0.5
REDDIT-B	73.7 ± 1.1	74.5 ± 0.8	75.7 ± 0.5	76.5 ± 0.4	76.3 ± 0.7	76.4 ± 0.6
REDDIT-M-5k	37.3 ± 1.2	40.2 ± 0.7	43.5 ± 0.7	44.0 ± 0.6	44.3 ± 0.4	44.2 ± 0.5

Table 7
Performance *w.r.t.* the random walk length on all datasets (in %).

Random walk length	1	2	3	4	5	6
PROTEINS	70.5 ± 0.8	71.1 ± 1.2	71.2 ± 0.7	71.0 ± 1.0	71.0 ± 1.5	70.0 ± 0.9
DD	69.4 ± 1.3	71.2 ± 0.7	71.4 ± 0.8	71.6 ± 0.7	71.2 ± 0.6	71.0 ± 0.8
IMDB-B	69.5 ± 0.9	71.3 ± 1.0	72.3 ± 1.1	72.0 ± 1.2	71.5 ± 1.3	71.7 ± 1.1
IMDB-M	43.3 ± 0.6	44.4 ± 0.5	44.5 ± 0.6	44.2 ± 0.5	43.8 ± 0.7	44.0 ± 0.5
REDDIT-B	75.7 ± 0.8	76.6 ± 0.8	76.5 ± 0.4	76.6 ± 0.7	76.2 ± 0.8	75.6 ± 1.2
REDDIT-M-5k	41.8 ± 1.0	43.1 ± 0.7	44.0 ± 0.6	43.4 ± 0.8	43.6 ± 0.7	43.1 ± 0.9

The results of the model variants are summarized in Table 5. First, we can observe that our full model outperforms GraphSpa w/o A consistently, which indicates that the active learning strategy plays a vital role in our semi-supervised graph classification, thus implying the effectiveness of our hybrid selection strategy. Second, the full model also outperforms both GraphSpa w/o L and GraphSpa w/o G, showing that both the local selection strategy and global selection strategy are indispensable for improving the performance. Moreover, GraphSpa w/o L performs better than GraphSpa w/o G, which demonstrates the superiority of the uncertainty in the global selection strategy. Third, with contrastive loss, GraphSpa w/o A outperforms GNN-Sup and the full model outperforms GraphSpa w/o C, both of these comparisons support our assumption that contrastive loss may be beneficial in semi-supervised scenarios, which aligns with our expectations.

4.4. Parameter analysis

Here we study how the performance of GraphSpa varies with different parameter settings. Specifically, we investigate the impact of the embedding dimensions of hidden layers d in Table 6, the random walk length p in Table 7 and different hybrid strategies in Table 8.

Impact of the embedding dimensions. We begin by examining the effect of the embedding dimensions of hidden layers d. We hypothesize that increasing the embedding dimensions would enhance the model's capacity and, thus, its performance. We fix all other parameters to their optimal values and vary d in the range of $\{8, 16, 32, 64, 128\}$. Our observations indicate that enlarging the embedding size generally results in performance improvements until a point of saturation is reached. The model exhibits a certain level of fluctuation, or even a decline, when using particularly large embedding dimensions. The possible reason is that the model has reached saturation, and further increasing the dimension may lead to underfitting.

Table 8
Impact of three hybrid fusion strategies on three datasets (in %).

Strategies	PROTEINS	DD	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M-5k
Intersection	71.2 ± 0.7	71.4 ± 0.8	72.3 ± 1.1	44.5 ± 0.6	76.5 ± 0.4	44.0 ± 0.6
Union	71.6 ± 1.1	71.0 ± 0.9	72.0 ± 0.8	43.8 ± 0.8	75.9 ± 0.5	43.7 ± 0.7
Attention	70.7 ± 1.6	71.5 ± 0.8	72.6 ± 1.2	44.2 ± 1.3	76.1 ± 0.5	43.9 ± 0.6

Table 9
Comparisons of run time (second) needed per active learning cycle.

Methods	PROTEINS	DD	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M-5k
MCDAL	44.4	62.0	44.3	56.4	34.3	180.6
ASGNN	56.3	78.1	36.0	63.8	72.6	312.3
GraphSpa	27.1	51.0	26.4	46.9	32.6	206.3

Impact of the random walk length. We conduct further investigation on the impact of the random walk length p in the local selection strategy. By varying p in the range of $\{1, 2, 3, 4, 5, 6\}$ while keeping the other parameters constant, we observe that increasing p improves the model's performance, particularly when p is small. This suggests that our random walk kernel can effectively detect more valid substructures with larger lengths, thereby enhancing the learning of graph topology. However, if p is too large, it may lead to a decrease in performance. This may be due to the fact that excessively long random walks are less stable in distinguishing graph similarities.

Impact of different hybrid fusion strategies. We finally investigate the impact of different hybrid fusion strategies for our approach. As illustrated in Table 8, our results indicate that the performance of our approach is not significantly affected by different hybrid fusion strategies, suggesting the robustness of our fusion selection strategy. Interestingly, the results of the "Union" strategy were lower than those of the other two strategies. This may be due to the selected samples not comprehensively considering the agreement of both strategies, leading to biased sample selection.

4.5. Runtime analysis

Here, we compare our proposed GraphSpa method with the two latest active learning methods. We test the runtime of the selection strategy for each cycle of active learning to further demonstrate the efficiency of our proposed strategy. As shown in Table 9, we can observe that the runtime of our active learning selection strategy is the shortest on almost all datasets, fully illustrating the robustness and efficiency of our selection strategy. This makes it more suitable for many practical applications, especially in fields prioritizing efficiency.

4.6. Case study

We analyze the learning curves and convergence depicted in Fig. 4. We take the DD as an example, and compare our proposed GraphSpa with the representative baseline *Entropy*, which is widely considered as an uncertainty-based baseline. It chooses uncertain samples with the greatest entropy in terms of predicted class probabilities. As can be seen, both methods achieved a significant reduction in train and test losses within only a few iterations and eventually converged well.

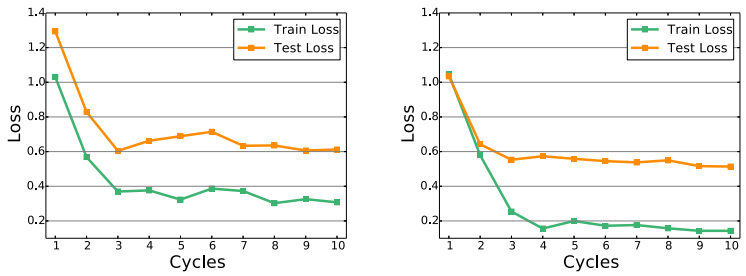

Fig. 4. Loss *w.r.t.* two selection strategies on DD.

This suggests that selecting samples via active learning can indeed provide abundant supervision signals to guide the gradient optimization effectively. Additionally, the losses of GraphSpa decrease more steadily than those of the baseline for each cycle, indicating that GraphSpa may choose more relevant graphs beneficial to model training in each cycle. This case further demonstrates the superiority of our hybrid fusion selection strategy, which considers both local similarity and global semantic structure.

5. Discussion

5.1. Potential applicability

Our proposed active learning strategy can be highly applicable in real-life use cases for graph-level classification, offering several advantages in various domains as follows:

- Resource Allocation: In resource-constrained environments, our framework helps optimize the allocation of labeling resources. It ensures that limited resources are spent on the most critical and informative graphs, making it a valuable tool for organizations with budget constraints.
- Medicine and Biology: Our framework is instrumental in advancing bioinformatics and medical research, offering invaluable contributions to fields such as protein-protein interaction prediction and drug discovery. It assists in prioritizing experiments or data collection for the most promising candidates, reducing experimental costs.
- Natural Language Processing (NLP): In NLP tasks that involve graph representations, our proposed framework aids in document classification, entity recognition, or relation extraction by prioritizing the labeling of documents or entities that are most informative for the task.
- Environmental Monitoring: Our framework can be applied to environmental data analysis, such as ecosystem modeling or climate forecasting, by selecting the most critical data points or sensor readings for labeling to improve predictive accuracy.

5.2. Potential limitation

On the one hand, our active learning strategy involves using graph kernel techniques to compute the similarity between graphs. However, this may pose certain limitations when dealing with extremely large-scale graph data in practical deployments. In the future, we can explore the use of learnable graph kernel techniques to flexibly model the similarity computation between graphs, effectively increasing the scalability of our strategy. On the other hand, our active learning strategy requires calculating the probability prediction differences of graph samples between adjacent cycles, which could potentially increase the computational cost of the model. Additionally, besides the active learning strategy, we currently employ relatively common contrastive learning techniques to make the most of the abundant unlabeled graph data present in real-world applications. However, this approach may not fully and effectively extract the inherent semantics of the data. In future work, we can explore more sophisticated techniques to uncover more comprehensive semantic information from graph data, such as large-scale pretraining.

6. Conclusion

This paper tackles the task of semi-supervised graph-level classification under limited labeling budgets, which is a practical yet under-explored problem. To address this challenge, we propose GraphSpa, an effective approach that actively selects informative graphs for subsequent training using our hybrid fusion selection strategy that combines local similarity and global semantic structure. Furthermore, we introduce a novel semi-supervised active learning framework that incorporates graph contrastive learning into active learning. Our extensive experiments on a range of well-known benchmark datasets demonstrate the effectiveness of our proposed GraphSpa.

Going forward, we plan to extend our method to real-world applications such as molecular conformation generation and protein function prediction. We also aim to enhance our approach by incorporating graph similarity learning and advanced bootstrapping theories to improve sample selection.

CRediT authorship contribution statement

Wei Ju: Conceptualization, Funding acquisition, Writing – original draft. **Zhengyang Mao:** Software, Writing – review & editing. **Ziyue Qiao:** Software, Writing – review & editing. **Yifang Qin:** Software, Writing – review & editing. **Siyu Yi:** Software, Writing – review & editing. **Zhiping Xiao:** Writing – review & editing. **Xiao Luo:** Conceptualization, Methodology, Writing – review & editing. **Yanjie Fu:** Conceptualization, Methodology. **Ming Zhang:** Conceptualization, Funding acquisition, Project administration.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

This paper is partially supported by the National Natural Science Foundation of China with Grant (NSFC Grant No. 62306014 and No. 62276002) as well as the China Postdoctoral Science Foundation with Grant No. 2023M730057.

References

[1] W. Ju, Z. Fang, Y. Gu, Z. Liu, Q. Long, Z. Qiao, Y. Qin, J. Shen, F. Sun, Z. Xiao, et al., A comprehensive survey on deep graph representation learning, 2023, arXiv preprint arXiv:2304.05055.

[2] Z. Hao, C. Lu, Z. Huang, H. Wang, Z. Hu, Q. Liu, E. Chen, C. Lee, ASGN: An active semi-supervised graph neural network for molecular property prediction, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 731–752.

[3] R. Kojima, S. Ishida, M. Ohta, H. Iwata, T. Honma, Y. Okuno, kGCN: a graph-based deep learning framework for chemical structures, J. Cheminform. 12 (2020) 1–10.

[4] Z. Ying, J. You, C. Morris, X. Ren, W. Hamilton, J. Leskovec, Hierarchical graph representation learning with differentiable pooling, Adv. Neural Inf. Process. Syst. 31 (2018).

[5] W. Ju, J. Yang, M. Qu, W. Song, J. Shen, M. Zhang, Kgnn: Harnessing kernel-based networks for semi-supervised graph classification, in: Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, 2022, pp. 421–429.

[6] W. Ju, X. Luo, M. Qu, Y. Wang, C. Chen, M. Deng, X.-S. Hua, M. Zhang, TGNN: A joint semi-supervised framework for graph-level classification, 2023, arXiv preprint arXiv:2304.11688.

[7] N.M. Kriege, F.D. Johansson, C. Morris, A survey on graph kernels, Appl. Netw. Sci. 5 (1) (2020) 1–42.

[8] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: International Conference on Learning Representations, 2017.

[9] R. Zheng, W. Chen, G. Feng, Semi-supervised node classification via adaptive graph smoothing networks, Pattern Recognit. 124 (2022) 108492.

[10] X. Lin, C. Zhou, J. Wu, H. Yang, H. Wang, Y. Cao, B. Wang, Exploratory adversarial attacks on graph neural networks for semi-supervised node classification, Pattern Recognit. 133 (2023) 109042.

[11] W. Ju, Y. Qin, Z. Qiao, X. Luo, Y. Wang, Y. Fu, M. Zhang, Kernel-based substructure exploration for next POI recommendation, in: 2022 IEEE International Conference on Data Mining, ICDM, IEEE, 2022, pp. 221–230.

[12] X. Luo, Y. Zhao, Y. Qin, W. Ju, M. Zhang, Towards semi-supervised universal graph classification, IEEE Trans. Knowl. Data Eng. (2023).

[13] Q. Sun, J. Li, H. Peng, J. Wu, Y. Ning, P.S. Yu, L. He, Sugar: Subgraph neural network with reinforcement pooling and self-supervised mutual information mechanism, in: Proceedings of the Web Conference, 2021.

[14] E. Engel, R.M. Dreizler, Density Functional Theory, Springer, 2013.

[15] M. Gao, Z. Zhang, G. Yu, S.Ö. Arık, L.S. Davis, T. Pfister, Consistency-based semi-supervised active learning: Towards minimizing labeling cost, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16, Springer, 2020, pp. 510–526.

[16] S. Huang, T. Wang, H. Xiong, J. Huan, D. Dou, Semi-supervised active learning with temporal output discrepancy, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 3447–3456.

[17] J. Guo, H. Shi, Y. Kang, K. Kuang, S. Tang, Z. Jiang, C. Sun, F. Wu, Y. Zhuang, Semi-supervised active learning for semi-supervised models: Exploit adversarial examples with graph-based virtual labels, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 2896–2905.

[18] J. Yuan, X. Luo, Y. Qin, Y. Zhao, W. Ju, M. Zhang, Learning on graphs under label noise, in: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2023, pp. 1–5.

[19] X. Luo, W. Ju, M. Qu, Y. Gu, C. Chen, M. Deng, X.-S. Hua, M. Zhang, Clear: Cluster-enhanced contrast for self-supervised graph representation learning, IEEE Trans. Neural Netw. Learn. Syst. (2022).

[20] G. Wang, J.-N. Hwang, C. Rose, F. Wallace, Uncertainty-based active learning via sparse modeling for image classification, IEEE Trans. Image Process. 28 (1) (2018) 316–329.

[21] A.J. Joshi, F. Porikli, N. Papanikolopoulos, Multi-class active learning for image classification, in: 2009 Ieee Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 2372–2379.

[22] J.W. Cho, D.-J. Kim, Y. Jung, I.S. Kweon, Mcdal: Maximum classifier discrepancy for active learning, IEEE Trans. Neural Netw. Learn. Syst. (2022).

[23] O. Sener, S. Savarese, Active learning for convolutional neural networks: A core-set approach, in: International Conference on Learning Representations, 2018.

[24] R. Wang, X.-Z. Wang, S. Kwong, C. Xu, Incorporating diversity and informativeness in multiple-instance active learning, IEEE Trans. Fuzzy Syst. 25 (6) (2017) 1460–1475.

[25] S. Agarwal, H. Arora, S. Anand, C. Arora, Contextual diversity for active learning, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16, Springer, 2020, pp. 137–153.

[26] Y. Yan, G.M. Fung, R. Rosales, J.G. Dy, Active learning from crowds, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), 2011, pp. 1161–1168.

[27] W. Cai, Y. Zhang, J. Zhou, Maximizing expected model change for active learning in regression, in: 2013 IEEE 13th International Conference on Data Mining, IEEE, 2013, pp. 51–60.

[28] A. Freytag, E. Rodner, J. Denzler, Selecting influential examples: Active learning with expected model output changes, in: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13, Springer, 2014, pp. 562–577.

[29] Y. Grandvalet, Y. Bengio, Semi-supervised learning by entropy minimization, Adv. Neural Inf. Process. Syst. 17 (2004).

[30] Q. Xie, M.-T. Luong, E. Hovy, Q.V. Le, Self-training with noisy student improves imagenet classification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10687–10698.

[31] S. Laine, T. Aila, Temporal ensembling for semi-supervised learning, in: International Conference on Learning Representations, 2017.

[32] A. Tarvainen, H. Valpola, Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, Adv. Neural Inf. Process. Syst. 30 (2017).

[33] F.-Y. Sun, J. Hoffmann, V. Verma, J. Tang, Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization, in: International Conference on Learning Representations, 2020.

[34] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, Y. Shen, Graph contrastive learning with augmentations, Adv. Neural Inf. Process. Syst. 33 (2020) 5812–5823.

[35] Y. You, T. Chen, Y. Shen, Z. Wang, Graph contrastive learning automated, in: International Conference on Machine Learning, PMLR, 2021, pp. 12121–12132.

[36] X. Luo, W. Ju, M. Qu, C. Chen, M. Deng, X.-S. Hua, M. Zhang, Dualgraph: Improving semi-supervised graph classification via dual contrastive learning, in: 2022 IEEE 38th International Conference on Data Engineering, ICDE, IEEE, 2022, pp. 699–712.

[37] A. Abraham, L. Dreyfus-Schmidt, Rebuilding trust in active learning with actionable metrics, in: 2020 International Conference on Data Mining Workshops, ICDMW, IEEE, 2020, pp. 836–843.

[38] D. Bahri, H. Jiang, T. Schuster, A. Rostamizadeh, Is margin all you need? An extensive empirical study of active learning on tabular data, 2022, arXiv preprint arXiv:2210.03822.

[39] J. Gilmer, S.S. Schoenholz, P.F. Riley, O. Vinyals, G.E. Dahl, Neural message passing for quantum chemistry, in: International Conference on Machine Learning, PMLR, 2017, pp. 1263–1272.

[40] K. Xu, W. Hu, J. Leskovec, S. Jegelka, How powerful are graph neural networks? in: International Conference on Learning Representations, 2019.

[41] B. Radunovic, J.-Y. Le Boudec, A unified framework for max-min and min-max fairness with applications, IEEE/ACM Trans. Netw. 15 (5) (2007) 1073–1083.

[42] S.V.N. Vishwanathan, N.N. Schraudolph, R. Kondor, K.M. Borgwardt, Graph kernels, J. Mach. Learn. Res. 11 (2010) 1201–1242.

[43] A.v.d. Oord, Y. Li, O. Vinyals, Representation learning with contrastive predictive coding, 2018, arXiv preprint arXiv:1807.03748.

[44] C. Morris, N.M. Kriege, F. Bause, K. Kersting, P. Mutzel, M. Neumann, Tudataset: A collection of benchmark datasets for learning with graphs, 2020, arXiv preprint arXiv:2007.08663.

[45] N. Shervashidze, S. Vishwanathan, T. Petri, K. Mehlhorn, K. Borgwardt, Efficient graphlet kernels for large graph comparison, in: Artificial Intelligence and Statistics, PMLR, 2009, pp. 488–495.

[46] K.M. Borgwardt, H.-P. Kriegel, Shortest-path kernels on graphs, in: Fifth IEEE International Conference on Data Mining, ICDM'05, IEEE, 2005, pp. 8–pp.

[47] N. Shervashidze, P. Schweitzer, E.J. Van Leeuwen, K. Mehlhorn, K.M. Borgwardt, Weisfeiler-lehman graph kernels, J. Mach. Learn. Res. 12 (9) (2011).

[48] P. Yanardag, S. Vishwanathan, Deep graph kernels, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015, pp. 1365–1374.

[49] B. Adhikari, Y. Zhang, N. Ramakrishnan, B.A. Prakash, Sub2vec: Feature learning for subgraphs, in: Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2018.

[50] A. Narayanan, M. Chandramohan, R. Venkatesan, L. Chen, Y. Liu, S. Jaiswal, graph2vec: Learning distributed representations of graphs, 2017, arXiv preprint arXiv:1707.05005.

[51] T. Miyato, S.-i. Maeda, M. Koyama, S. Ishii, Virtual adversarial training: a regularization method for supervised and semi-supervised learning, IEEE Trans. Pattern Anal. Mach. Intell. 41 (8) (2018) 1979–1993.

[52] W. Ju, X. Luo, Z. Ma, J. Yang, M. Deng, M. Zhang, GHNN: Graph Harmonic Neural Networks for semi-supervised graph-level classification, Neural Netw. 151 (2022) 70–79.

[53] J. Zhang, J. Katz-Samuels, R. Nowak, Galaxy: Graph-based active learning at the extreme, in: International Conference on Machine Learning, PMLR, 2022, pp. 26223–26238.

[54] Y. Xie, S. Lv, Y. Qian, C. Wen, J. Liang, Active and semi-supervised graph neural networks for graph classification, IEEE Trans. Big Data 8 (4) (2022) 920–932.

Wei Ju is currently a postdoc research fellow in Computer Science at Peking University. Prior to that, he received his Ph.D. degree in Computer Science from Peking University, Beijing, China, in 2022. He received the B.S. degree in Mathematics from Sichuan University, Sichuan, China, in 2017. His current research interests lie primarily in the area of machine learning on graphs including graph representation learning and graph neural networks, and interdisciplinary applications such as bioinformatics, drug

discovery, recommender systems, spatio-temporal analysis and knowledge graphs. He has published more than 30 papers in top-tier venues and has won the best paper finalist in IEEE ICDM 2022.

Zhengyang Mao is currently a graduate student at the School of Computer Science, Peking University. His research interests lie primarily in the area of machine learning with graph, including graph representation learning, data-imbalanced learning, and semi-supervised learning.

Ziyue Qiao is currently an Assistant Professor at the School of Computing and Information Technology, Great Bay University. Previously, he was postdoctoral fellow at The Hong Kong University of Science and Technology (Guangzhou) from 2022 to 2024. He received his Ph.D. degree in 2022 at the Computer Network Information Center, Chinese Academy of Sciences, and his B.S. degree in 2017 at Wuhan University, China. His research interests include data mining, graph learning, and natural language processing, with an emphasis on designing new algorithms for graph representation/transfer learning and academic data mining.

Yifang Qin is currently a graduate student in School of Computer Science, Peking University, Beijing, China. Prior to that, he received the B.S. degree in school of EECS, Peking University. His research interests include graph representation learning and recommender systems.

Siyu Yi is currently a Ph.D. candidate in statistics from Nankai University, Tianjin, China. She received the B.S. and M.S. degrees in Mathematics from Sichuan University, Sichuan, China, in 2017 and 2020, respectively. Her research interests focus on graph representation learning, design of experiments, and subsampling in big data.

Zhiping Xiao is currently a Ph.D. candidate in Computer Science at University of California, Los Angeles. She plans to graduate in year 2024. Her major is data mining, and her current research interests lie in the area of multi-modality social-media data analysis. She is also interested in other interdisciplinary applications.

Xiao Luo is a postdoctoral researcher in Department of Computer Science, University of California, Los Angeles, USA. Prior to that, he received the Ph.D. degree in School of Mathematical Sciences from Peking University, Beijing, China and the B.S. degree in Mathematics from Nanjing University, Nanjing, China, in 2017. His research interests includes machine learning on graphs, image retrieval, statistical models and bioinformatics.

Yanjie Fu is an associate professor in the School of Computing and AI at the Arizona State University. He received his Ph.D. degree from the Rutgers, the State University of New Jersey in 2016, the B.E. degree from the University of Science and Technology of China in 2008, and the M.E. degree from the Chinese Academy of Sciences in 2011. He has research experience in industry research labs, such as Microsoft Research Asia and IBM Thomas J. Watson Research Center. He has published prolifically in refereed journals and conference proceedings, such as IEEE TKDE, IEEE TMC, ACM TKDD, ACM SIGKDD, AAAI, IJCAI, VLDB, WWW, ACM SIGIR. His research has been recognized by: 1) two federal junior faculty awards: US NSF CAREER and NSF CRII awards; 2) five best paper (runner-up, finalist) awards, including ACM KDD18 Best Student Paper Finalist, IEEE ICDM14, 21, 22 Best Paper Finalist, ACM SIGSpatial20 Best Paper Runner-up; 3) three industrial awards: 2016 Microsoft Azure Research Award, 2022 Baidu Scholar global top Chinese young scholars in AI, 2021 Aminer.org AI 2000 Most Influential Scholar Award Honorable Mention in Data Mining; 4) several other university-level awards: Reach the Stars Award, University System Research Board Award and University Interdisciplinary Research Award. He was chosen for the nation's early career engineers by the National Academy of Engineering 2023 Grainger Foundation Frontiers of Engineering Symposium. He is broadly interested in data mining, machine learning, and their interdisciplinary applications. His research aims to develop robust machine intelligence with imperfect and complex data by building tools to address framework, algorithmic, data, and computing challenges. His recent focuses are spatial-temporal AI, graph learning, reinforcement learning, learning with unlabeled data, stream learning and distribution drift. He currently serves as an Associate Editor of ACM Transactions on Knowledge Discovery from Data and Mathematics. He is a senior member of ACM and IEEE.

Ming Zhang received her B.S., M.S. and Ph.D. degrees in Computer Science from Peking University respectively. She is a full professor at the School of Computer Science, Peking University. Prof. Zhang is a member of Advisory Committee of Ministry of Education in China and the Chair of ACM SIGCSE China. She is one of the fifteen members of ACM/IEEE CC2020 Steering Committee. She has published more than 200 research papers on Text Mining and Machine Learning in the top journals and conferences. She won the best paper of ICML 2014 and best paper nominee of WWW 2016. Prof. Zhang is the leading author of several textbooks on Data Structures and Algorithms in Chinese, and the corresponding course is awarded as the National Elaborate Course, National Boutique Resource Sharing Course, National Fine-designed Online Course, National First-Class Undergraduate Course by MOE China.