Perturbation-Based Graph Active Learning for Semi-Supervised Belief Representation Learning

Dachun Sun, Jinning Li, Xinyi Liu, You Lyu, Hongjue Zhao, Denizhan Kara, and Tarek Abdelzaher
Department of Computer Science
University of Illinois Urbana-Champaign, Urbana IL 61801, USA
{dsun18, jinning4, liu323, youlyu2, hongjue2, kara4, zaher}@illinois.edu

***Abstract*—This paper addresses the problem of optimizing the allocation of labeling resources to enhance the performance of *semi-supervised* belief representation learning in social networks. The objective is to strategically identify valuable nodes in social media graphs that are worth labeling within a constrained budget to maximize downstream learning task performance. Despite progress in unsupervised and semi-supervised methods for belief and ideology representation learning on social networks, the scarcity of high-quality labeled social data continues to pose a significant challenge. Therefore, allocating labeling efforts judiciously becomes critical in scenarios with limited resources for labeling. This paper introduces a perturbation-based active learning strategy inspired by graph augmentation, PerbALGraph, which progressively selects nodes for labeling using an automatic estimator, thereby eliminating the need for human guidance. This estimator is based on the principle that nodes in the network that exhibit heightened sensitivity to changes in structural features are better candidates for labeling. We design the estimator to be model-agnostic and application-independent and to score candidates under a set of designed graph perturbations. Extensive experiments on six real-world social media datasets demonstrate the superior performance and robustness of our proposed method compared to existing active learning approaches.**

***Index Terms*—Graph Active Learning, Semi-Supervised Graph Representation Learning, Social Networks, Belief Embedding.**

I. Introduction

As social media networks become ever more central to our society, individuals' online behaviors increasingly reveal their beliefs, such as stances, viewpoints, and ideological preferences. To capture these insights, *belief representation learning* has been used to uncover latent graph structures that correlate with specific beliefs based on observable online behaviors on social media [1]–[5]. State-of-the-art methods deploy Graph Neural Networks (GNNs) to derive unsupervised belief representations from online behavioral graphs (e.g., who shares or posts what) [3], [5], [6], enabling advanced applications like classifying users and posts by their beliefs [2], forecasting stances on various topics [7], and identifying polarization within belief spaces [8].

While unsupervised GNN-based methods offer several advantages, they encounter limitations in specific real-world tasks, such as when graphs are sparse. If a small number of nodes are labeled for the task, it may greatly improve downstream inference quality. In social networks, for instance, the inherent power law distribution of node connectivity creates challenges in accurately inferring the beliefs of users within isolated or sparse components, where the structural inductive biases learned through unsupervised methods prove ineffective. Additionally, supervision becomes essential in belief representation learning tasks for aligning the computed latent belief representation with the definitions of human-interpretable ideologies. For example, in discussions surrounding geopolitical events, supervision is necessary to categorize users and posts along a user-defined interpretable axis, such as Conservatism versus Liberalism, ensuring that the computed latent representation aligns correctly with the definitions of these particular human-interpretable ideological stances.

Despite the vast amount of unlabeled data in social networks and progress in GNNs, acquiring labeled data for social applications remains challenging due to limited budgets and resources. Thus, allocating labeling efforts strategically towards valuable nodes within a constrained budget is essential to enhance performance. Active learning (AL) [9] aims to identify the most informative instances for labeling. To this end, various strategies have been proposed, including centrality and uncertainty-based methods [10], [11], heuristic approaches such as AGE [12] and ANRMAB [13], and reinforcement learning-based methods like GPA [14]. However, when applied to social applications, these baselines extract limited information from the iteration process and the graph structure. For instance, they are not tuned for the unique characteristics of bipartite social graphs, where nodes can be of different types and the connectivity patterns differ, making them less effective at managing labeling budgets for social applications.

We introduce a novel approach called PerbALGraph to extract deeper signals from the iterative process and graph structure by leveraging input graph variations through designed perturbations. These perturbations, tailored specifically for graph-structured data, include techniques such as edge dropping, noisy edge addition, and path dropping. By analyzing the learning model's inferences on these perturbed graphs and the differences in node properties across various perturbations, we derive more robust signals that work well in selecting valuable nodes for labeling in social graphs.

To evaluate the effectiveness of our proposed AL strategy, we conduct experiments on two types of GNN-based encoders and a node classification task, specifically ideology identification in social media data. We also empirically examine the selected nodes in the social media data to reveal more insights about the approach in specific case studies. Our experiment results show that the proposed approach outperforms existing methods in classification accuracy and labeling efficiency. This suggests that it is a promising direction for optimizing the allocation of the labeling budget in semi-supervised belief representation learning and social network applications. To sum up, our contributions are:

- We propose an AL algorithm that quantifies the value of a node through performance variance (instability and sensitivity) induced by graph perturbations, which selects more valuable nodes to label from social network graphs, facilitating better downstream inference task performance.
- We specifically study the application of our graph AL algorithm to the semi-supervised belief representation learning problem.
- We conduct extensive experiments on six social media datasets and include several case studies. The results demonstrate the superiority of the proposed PerbALGraph method at improving downstream inference quality for a given labeling budget.

The paper is structured as follows. Section II states the problem definition. Section III introduces our proposed method, followed by the experiment results and analysis in Section IV. We review the literature on graph learning, active learning, and social media data mining in Section V. We discuss the advantages, limitations, and future work in Section VI and conclude with a summary in Section VII.

II. Problem Definition

We consider a bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}) = (\mathcal{V}_U \cup \mathcal{V}_P, \mathcal{E})$ with $|\mathcal{V}| = n$, the number of belief-related labels c, adjacency matrix $\mathbf{A} \in \{0,1\}^{n\times n}$ ($\mathbf{A} \in \mathbb{R}^{n\times n}$ if the graph is weighted), and node features $\mathbf{X} = \left[\mathbf{x}_1^\top, \mathbf{x}_2^\top, ..., \mathbf{x}_n^\top\right]$, where $x_i \in \mathbb{R}^d$ and d is the feature dimension. The set of vertices, $\mathcal{V}$, is the union of users ($\mathcal{V}_U$) and consolidated posts ($\mathcal{V}_P$, that we shall call *assertions*, collapsing together messages that convey the same idea). Assertion node set $\mathcal{V}_P$ is split into training set $\mathcal{V}_{\text{train}}$, validation set $\mathcal{V}_{\text{val}}$, and testing set $\mathcal{V}_{\text{test}}$. Node features can be derived from assertion text, labels, or graph connectivity.

Given a budget B, a labeling oracle that can assign an assertion v_i to the belief label $y_i \in \{0,1\}^c$, and a set of initially labeled nodes $\mathcal{V}_L \subseteq \mathcal{V}_{\text{train}}$ ($n_L = |\mathcal{V}_L|$), the objective is to optimize the performance of the belief representation learning model by designing an active learning selection strategy to select $(B - n_L)$ nodes from the unlabeled candidate node set $\mathcal{V}_C$ for the oracle to label and append to $\mathcal{V}_L$. This optimization problem is formulated as maximizing the likelihood of correct predictions over the test data, expressed as follows:

$$\underset{\mathcal{V}_L:|\mathcal{V}_L|=B}{\arg\min}\ \mathbb{E}_{v_i\in\mathcal{V}_{\text{test}}}\left[\mathcal{L}\left(y_i, P(\hat{y}_i|f_\phi(G;\mathcal{V}_L))\right)\right] \tag{1}$$

where $P(\hat{y}_i|f_\phi)$ is the predicted label distribution of node v_i by the belief representation learning model f_ϕ (e.g., GNN).

We follow the setting of a pool-based active learning, applied to the input graph. Initially, the only known labels are those available for the set, $\mathcal{V}_L, \{y_i\}_{i\in\mathcal{V}_L}$. Our method then identifies a query node set $\mathcal{V}_Q$ from $\mathcal{V}_C$ and queries the oracle for labels. These node sets are correspondingly updated

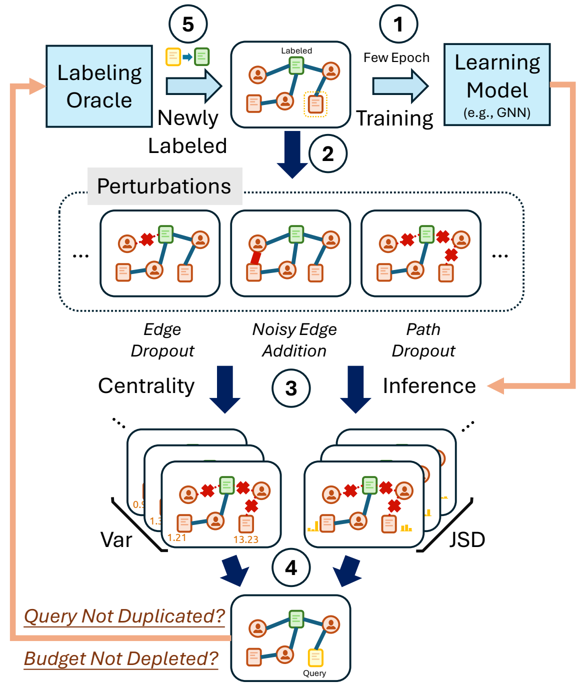


Fig. 1: Overview of the PerbALGraph framework.

($\mathcal{V}_L \leftarrow \mathcal{V}_L \cup \mathcal{V}_Q$, $\mathcal{V}_C \leftarrow \mathcal{V}_C \backslash \mathcal{V}_Q$) until the labeling budget is depleted. As a hyperparameter, the size $b = |\mathcal{V}_Q|$ decides how many nodes to label in each iteration.

III. Methodology

A. Perturbation-based Graph Active-Learning

Many existing methods [12]–[14] focus on extracting the informativeness or representativeness of a node on the original graph $\mathcal{G}$ with an internal learning model. Predicted label distribution on a node tends to be the most informative. Inspired by work on graph data augmentation and graph contrastive learning [15], we propose to extract more signals from the learning model and the graph structure using input graph variations by applying perturbations designed for graph-structured data.

1) Graph Perturbation: As a data augmentation method mainly to facilitate contrastive learning on graphs [16], we apply the following augmentations to elicit the noteworthy node candidates from the learning model.

- *Edge dropping:* Randomly dropping edges from the adjacency matrix with probability p using samples from a Bernoulli distribution.
- *Add random edge:* Randomly adding edges to the adjacency matrix.
- *Path dropping:* Dropping edges from the adjacency matrix based on random walks.

After performing these operations a_e, a_m, and a_p times respectively, we obtain a set of perturbed graphs $\{\tilde{\mathcal{G}}_i\}_{i\in[1,\mathbf{a}]}$, where $\mathbf{a} = \sum a_*$. We do not employ node dropping because the underlying prior for it is that missing a vertex does not alter semantics [16]. Our method calculates several metrics for each unlabeled node and identifies a query node set.

2) Instability: Unlike pure entropy-based methods, we pay attention to the nodes that would switch the "side" predicted by the learning model under different perturbations. The key insight is that when a node has high instability, meaning that the perturbed graph structure greatly changes the learning model's predicted label distribution, it is highly suitable to be chosen as an anchor for labeling.

To elicit this signal from the learning model and the graph structure, we compute the predicted label distributions for unlabeled nodes in *each* perturbed graph, $\{\mathbf{P}_i\}_{v_i \in \mathcal{V}_C}$, where each $\mathbf{P}_i = \left[P\left(\hat{y}_i | f_\phi(\tilde{\mathcal{G}}_j; \mathcal{V}_L)\right)\right]_{j\in[1,\mathbf{a}]}$ consists of predicted label distributions of v_i evaluated by the learning model on each perturbed graph. To be more specific, each $\mathbf{P}_{i,j}$ is a label distribution of node v_i inferred by the learning model on graph $\tilde{\mathcal{G}}_j$. We define the **instability** of v_i,

$$\phi_{\text{instability}}(v_i) = \text{JSD}(\mathbf{P}_i) \tag{2}$$

$$\text{JSD}(\mathbf{P}_*) = H\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{P}_{*,i}\right) - \frac{1}{n}\sum_{i=1}^{n} H(\mathbf{P}_{*,i}) \tag{3}$$

where JSD is called the generalized Jensen-Shannon divergence and $H(P)$ is the Shannon entropy.

3) Sensitivity: Another key component is the sensitivity score. In social network analysis, many metrics to measure the centrality of a node have been proposed. We adopt betweenness centrality [17] and an eigenvector-based PageRank centrality [18]. PageRank centrality of v_i is calculated as

$$\mu_{\text{PageRank}}(v_i; \mathcal{G}) = \rho \sum_{j=1}^{n} \mathbf{A}_{ij} \frac{\mu_{\text{PageRank}}(v_j; \mathcal{G})}{\sum_{k=1}^{n} \mathbf{A}_{jk}} + \frac{1-\rho}{n} \tag{4}$$

and the betweenness centrality of v_i is calculated as

$$\mu_{\text{betweenness}}(v_i; \mathcal{G}) = \sum_{\substack{s,v,t \in \mathcal{V}_\mathcal{G} \\ s \neq v \neq t}} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{5}$$

where σ_{st} is the total number of shortest paths from s to t, and $\sigma_{st}(v)$ is the number of those paths that pass through v.

We also compute the centrality metrics for *each* perturbed graph to gain more information about the graph structure. We calculate the centrality metrics for each unlabeled node on all perturbed graphs, giving $\left\{\boldsymbol{\mu}_i^{\text{(centrality)}}\right\}_{v_i \in \mathcal{V}_C}$, where each $\boldsymbol{\mu}_i^{\text{(centrality)}} = [\mu_{\text{centrality}}(v_i; \mathcal{G}_j)]_{j\in[1,\mathbf{a}]}$ We define the **sensitivity** of a node v_i to centrality metric m as

$$\phi_{\text{sensitivity}}(v_i) = \text{Var}\left[\boldsymbol{\mu}_i^{\text{(m)}}\right] \tag{6}$$

Previous centrality-based methods [12], [13] focus on graph AL tasks in tree-like datasets, such as citation graphs. The goal of deriving sensitivity scores from centrality metrics is to acquire better signals from raw metrics. For example, a star-shaped graph with a central hub node can be compared to another graph with two large components connected by a single node. Both nodes exhibit high betweenness measures, but they can be distinguished under sensitivity scores with edge perturbation. High node sensitivity (the bridging node) is valued more because the hub node, benefiting from consistent signals from its neighbors, may not need labeling for downstream belief representation tasks.

4) Score Combination: Since different metrics above are of incomparable scale, we first convert them into percentiles as in [12]. Define $\text{perc}_m(v, \mathcal{V}_C)$ as the percentile of nodes in $\mathcal{V}_C$ that have smaller values than node v according to metric m. Then, the objective function of our proposed method to select nodes for labeling is:

$$V^* = \underset{\substack{V \subset \mathcal{V}_C, \\ |V|=b}}{\arg\max} \sum_{v \in V} \Big[\gamma_t \cdot \text{perc}_{\text{instability}}(v, \mathcal{V}_C) + (1-\gamma_t) \cdot \text{perc}_{\text{sensitivity}}(v, \mathcal{V}_C)\Big] \tag{7}$$

We follow the settings in [12] and [19] where γ_t is a random variable between 0 and 1 with a temporal dependence on AL iteration, and it is responsible for shifting the attention from the structure-based sensitivity score to the learning-and-uncertainty-based instability score. We sample $\gamma_t \sim \text{Beta}(1, \beta_t)$. During the AL iteration, we decrease β_t linearly so that the expectation of γ_t increases, aligning with the need for increasing attention on the instability score. We refer to the combined score as the performance variance.

5) De-duplication: In our design, once a batch of queries is proposed, it is also possible to verify whether the new candidates are similar to those labeled. Specific tasks may use text embeddings as the node features $\mathbf{X}$, or a separate matrix might carry additional information. The query feature is compared against the features of those labeled. If the similarity score (cosine similarity, Euclidean distance, etc.) is below a threshold θ_d, the query will be discarded and not selected again. Given the small budget, performing such de-duplication should be efficient and ensure better use of the labeling budget.

The summary of PerbALGraph is presented in Algorithm 1. We analyze the runtime complexity of our algorithm, assuming that we use an L-layer Graph Convolutional Network (GCN) [20] as the learning model. The main computation components are graph perturbations, learning model training and inferences, and centrality calculation. For the PageRank variant, the dominating computation is on learning model, bounded by $\mathcal{O}\left(\mathbf{a}Ld(|\mathcal{E}| + d|\mathcal{V}|)\right)$, and for the betweenness variant, the dominating component is the centrality measure calculation, bounded by $\mathcal{O}(\mathbf{a}|\mathcal{E}||\mathcal{V}|)$, where $\mathbf{a}$ is the number of perturbed graphs and d is the node feature dimension.

IV. Experiments

In this section, we evaluate two variants (PageRank and Betweenness) of the proposed PerbALGraph methods and nine baseline models across six real-world X (formerly Twitter) datasets, using the belief representation as the downstream task. The evaluation is conducted under five fixed labeling budgets (5, 10, 15, 20, and 30), allowing us to investigate the impact of budget size on downstream tasks. These are reasonable budget sizes for semi-supervision tasks and for a single annotator to complete quickly. The code of PerbALGraph is available at https://github.com/dsun9/PerbALGraph.

Algorithm 1: Perturbation-Based Graph AL

Input: Budget B, graph $\mathcal{G}$, initial labeled node set $\mathcal{V}_L$, candidate node set $\mathcal{V}_C$, query set size b, perturbation config. (a_e, a_m, a_p)
Output: Selected node set $\mathcal{V}_{\text{selected}}$

$\mathcal{V}_{\text{selected}} \leftarrow \mathcal{V}_L$;
$t \leftarrow 1$;
while *budget B not depleted* **do**
    Train learning model $f_\phi(\mathcal{G}; \mathcal{V}_L)$;
    Form perturbed graphs $\{\tilde{\mathcal{G}}_i\}_{i=1}^{a_e+a_m+a_p}$;
    Run inference to obtain $\{\mathbf{P}_i\}_{v_i \in \mathcal{V}_C}$;
    Obtain centralities $\{\boldsymbol{\mu}_i^{(\text{centrality})}\}_{v_i \in \mathcal{V}_C}$;
    Calculate scores via Eqs. 2 and 6;
    Sample $\gamma_t \sim \text{Beta}(1, \beta_t)$ and compute combined scores;
    Compute batch V^* via Eq. 7;
    De-duplicate V^* against $\mathcal{V}_{\text{selected}}$ with threshold δ_d;
    $\mathcal{V}_{\text{selected}} \leftarrow \mathcal{V}_{\text{selected}} \cup V^*$;
    $B \leftarrow B - b$;
    $t \leftarrow t + 1$;
    Decrease β_t;
end
return $\mathcal{V}_{selected}$;

We perform extensive experiments with different AL methods under a constrained labeling budget of 20 nodes. We show that the PerbALGraph outperforms most baseline AL methods by selecting more valuable nodes to label, increasing the downstream belief classification performance across all datasets. The selected nodes are examined in case studies and proven to be appropriate and better than those chosen by baseline methods. We also provide a comparative analysis of our method against other AL methods across a range of labeling budgets, demonstrating its overall effectiveness.

Since PerbALGraph involves creating multiple copies of the input graph, we also evaluate its computational efficiency and scalability. Finally, we conduct an ablation study to assess the effect of varying the number of perturbations on the quality of selected nodes based on the downstream task performance.

A. Experimental Setup

1) Datasets: We employ six real-world datasets collected from the X platform using a combination of keyword and hashtag filters. These datasets contain social media posts processed by identifying re-posts (previously called retweets) and merging them into unified "assertions" for the belief representation learning task. Below is a brief description of each dataset and the labels used.

- **Eurovision 2016**: This dataset was collected during the Eurovision Song Contest 2016, focusing primarily on the competition and the Ukrainian singer Susana Jamaladinova, known professionally as Jamala. The labels are either pro-Jamala or anti-Jamala. Pro-Jamala users supported her victory, while anti-Jamala users claimed the contest results were politically influenced.
- **Russia/Ukraine Conflict (Visual)**: Collected between May and November 2022, this dataset was created using pro-Russia and pro-Ukraine keyword filters. Example keywords include "#standwithputin", "Minsk Accord", "NATO", and "BRICS". This dataset consists of only visual assertions, the clustered media (images) attached under the social media posts using CLIP [21]. The labels are pro-Ukraine or pro-Russia.
- **EDCA (Philippines), Territorial Sovereignty (Philippines), Insurgent Separatism (Philippines), and Energy Issues (China)**: Collected in 2023 using keyword filters related to geopolitical topics involving the Philippines, the United States, China, and the South China Sea. The dataset is further segmented into distinct topics using the PIEClass model [22], a weakly supervised topic classification model.
  - **EDCA (Philippines)**: Related keywords are "Enhanced Defense Cooperation Agreement", "EDCA", and etc. Labels are pro- or anti-alliance.
  - **Territorial Sovereignty (Philippines)**: Related keywords are "territorial sovereignty" and etc. Labels are pro- or anti-West.
  - **Energy Issues (China)**: Related keywords are "energy", "manufacturing", and etc. Labels are pro- or anti-China.
  - **Insurgent Separatism (Philippines)**: Related keywords are "Separatism", "autonomy", and etc. Labels are pro- or anti-Philippines.

TABLE I: Dataset Statistics. The average degree represents the overall sparsity. Dataset names are abbreviated.

Dataset	#Posts	#Assertions	#Users	#Edges	Avg. Deg.
Eurovision	3371	537	992	3081	4.03
EDCA	5219	467	912	4916	7.13
Russia/Ukraine	17153	1582	1577	5519	3.49
Sovereignty	101340	4386	6807	98302	17.56
Energy Issues	5950	189	3047	5939	3.67
Separatism	5474	750	957	5137	6.02

Table I summarizes the statistics for these datasets.

2) Baselines: We compare our method against the following baseline methods:

- **Random**: This method randomly selects B nodes for annotation.
- **Centrality-based**: This method selects B nodes for annotation based on a specific centrality metric (e.g., degree, PageRank, or betweenness).
- **Entropy-based**: At each step, this method selects the node with the highest information entropy in its predicted label distribution, as generated by the classification Graph Neural Network (GNN).
- **AGE** [12]: This method selects the node with the highest score at each step, where the score is a step-sensitive combination of three factors: the node's label distribution entropy, a centrality metric, and its representativeness (i.e., the distance of the node's embedding from the nearest cluster center).
- **ANRMAB** [13]: This approach uses the same three heuristic factors as AGE but combines them dynamically via a multi-armed bandit framework.

TABLE II: Evaluation results for belief representation learning using GCN [20] and SGVGAE [5] trained in a semi-supervised fashion on AL-queried nodes **with a budget of 20 queries**. The number next to the dataset name represents the percentage of the budget relative to the number of candidates in the dataset.

Dataset	Eurovision [3.7%]				EDCA (Philippines) [4.3%]			
Belief Embedding	GCN		SGVGAE		GCN		SGVGAE	
AL Method	Acc. (%)	Macro F1 (%)	Acc. (%)	Macro F1 (%)	Acc. (%)	Macro F1 (%)	Acc. (%)	Macro F1 (%)
Random	$77.25^{\pm 7.14}$	$74.23^{\pm 8.05}$	$76.36^{\pm 5.22}$	$75.29^{\pm 4.87}$	$83.75^{\pm 2.03}$	$77.17^{\pm 3.45}$	$65.91^{\pm 14.01}$	$63.12^{\pm 14.93}$
Centrality (Degree)	$83.47^{\pm 1.77}$	$80.73^{\pm 1.76}$	$74.00^{\pm 4.82}$	$72.88^{\pm 4.58}$	$84.24^{\pm 0.59}$	$76.86^{\pm 1.37}$	$65.53^{\pm 13.89}$	$63.88^{\pm 13.72}$
Centrality (PageRank)	$82.40^{\pm 4.79}$	$80.16^{\pm 4.73}$	$75.09^{\pm 5.32}$	$73.92^{\pm 5.12}$	$80.36^{\pm 1.44}$	$74.88^{\pm 1.56}$	$71.21^{\pm 13.23}$	$69.67^{\pm 13.14}$
Centrality (Betweenness)	$82.13^{\pm 3.58}$	$78.00^{\pm 3.60}$	$76.00^{\pm 5.91}$	$74.81^{\pm 5.78}$	$83.07^{\pm 3.08}$	$74.18^{\pm 7.86}$	$67.80^{\pm 11.80}$	$65.70^{\pm 11.82}$
Entropy	$78.39^{\pm 5.28}$	$73.36^{\pm 6.09}$	$76.48^{\pm 5.10}$	$75.31^{\pm 4.81}$	$83.57^{\pm 3.00}$	$78.44^{\pm 3.38}$	$61.36^{\pm 12.96}$	$58.20^{\pm 13.37}$
AGE	$79.07^{\pm 4.83}$	$76.31^{\pm 4.77}$	$76.00^{\pm 5.77}$	$74.90^{\pm 5.48}$	$84.99^{\pm 2.86}$	**$80.81^{\pm 2.67}$**	$61.36^{\pm 12.96}$	$58.20^{\pm 13.37}$
ANRMAB	$79.28^{\pm 6.70}$	$76.06^{\pm 7.08}$	$77.00^{\pm 5.55}$	$75.83^{\pm 5.28}$	$79.32^{\pm 4.61}$	$72.95^{\pm 4.52}$	$61.36^{\pm 12.96}$	$58.20^{\pm 13.37}$
GPA	$75.70^{\pm 6.62}$	$73.74^{\pm 5.94}$	$77.26^{\pm 4.86}$	$76.08^{\pm 4.58}$	$80.67^{\pm 2.24}$	$73.30^{\pm 2.08}$	$63.18^{\pm 12.80}$	$61.50^{\pm 12.62}$
RIM	$70.00^{\pm 0.00}$	$41.18^{\pm 0.00}$	$77.00^{\pm 4.63}$	$75.95^{\pm 4.32}$	$77.12^{\pm 4.25}$	$68.56^{\pm 3.85}$	$65.53^{\pm 13.89}$	$63.88^{\pm 13.72}$
Ours (PageRank)	**$85.14^{\pm 5.64}$**	**$81.71^{\pm 6.34}$**	**$81.33^{\pm 6.43}$**	**$79.98^{\pm 6.40}$**	**$86.04^{\pm 2.76}$**	$79.84^{\pm 4.66}$	**$73.86^{\pm 11.25}$**	**$71.21^{\pm 12.11}$**
Ours (Betweenness)	$82.86^{\pm 4.14}$	$78.57^{\pm 5.94}$	$79.50^{\pm 6.61}$	$78.38^{\pm 6.30}$	$79.55^{\pm 4.15}$	$72.36^{\pm 6.01}$	**$73.86^{\pm 11.25}$**	**$71.21^{\pm 12.11}$**

Dataset	Russia/Ukraine Conflict (Visual) [1.3%]				Territorial Sovereignty (Philippines) [0.5%]			
Belief Embedding	GCN		SGVGAE		GCN		SGVGAE	
AL Method	Acc. (%)	Macro F1 (%)	Acc. (%)	Macro F1 (%)	Acc. (%)	Macro F1 (%)	Acc. (%)	Macro F1 (%)
Random	$64.06^{\pm 2.76}$	$53.67^{\pm 9.33}$	$80.71^{\pm 2.79}$	$80.50^{\pm 2.87}$	$83.11^{\pm 0.00}$	$46.69^{\pm 0.00}$	$69.41^{\pm 11.49}$	$57.11^{\pm 8.23}$
Centrality (Degree)	$67.27^{\pm 0.95}$	$55.80^{\pm 1.87}$	$81.25^{\pm 2.89}$	$81.03^{\pm 2.98}$	$82.88^{\pm 0.00}$	$45.32^{\pm 0.00}$	$66.77^{\pm 12.42}$	$56.15^{\pm 7.82}$
Centrality (PageRank)	$59.89^{\pm 10.40}$	$57.31^{\pm 12.17}$	$80.24^{\pm 2.80}$	$80.02^{\pm 2.92}$	$82.88^{\pm 0.00}$	$45.32^{\pm 0.00}$	$69.00^{\pm 11.70}$	$55.10^{\pm 6.87}$
Centrality (Betweenness)	$67.58^{\pm 9.76}$	$65.27^{\pm 8.70}$	$80.68^{\pm 2.99}$	$80.51^{\pm 3.04}$	$82.88^{\pm 0.00}$	$45.32^{\pm 0.00}$	$72.98^{\pm 9.79}$	$58.51^{\pm 8.56}$
Entropy	$65.83^{\pm 1.89}$	$55.58^{\pm 5.62}$	$80.64^{\pm 2.30}$	$80.37^{\pm 2.44}$	**$83.21^{\pm 0.14}$**	$47.29^{\pm 0.81}$	$71.53^{\pm 10.51}$	$58.66^{\pm 8.00}$
AGE	$60.24^{\pm 0.11}$	$40.14^{\pm 0.28}$	$79.87^{\pm 4.20}$	$79.62^{\pm 4.30}$	$77.63^{\pm 7.50}$	$46.58^{\pm 1.90}$	$69.60^{\pm 10.64}$	$58.08^{\pm 7.88}$
ANRMAB	$65.09^{\pm 3.31}$	$58.33^{\pm 5.19}$	$80.65^{\pm 3.91}$	$80.39^{\pm 4.03}$	$82.88^{\pm 0.00}$	$45.32^{\pm 0.00}$	$69.07^{\pm 11.54}$	$56.48^{\pm 8.10}$
GPA	$61.10^{\pm 7.85}$	$56.50^{\pm 9.74}$	$81.20^{\pm 3.84}$	$80.96^{\pm 3.96}$	$67.83^{\pm 11.18}$	$51.67^{\pm 3.58}$	$67.67^{\pm 10.92}$	$55.89^{\pm 7.14}$
RIM	$43.18^{\pm 0.00}$	$34.52^{\pm 0.00}$	$80.28^{\pm 3.35}$	$80.04^{\pm 3.43}$	$82.88^{\pm 0.00}$	$45.32^{\pm 0.00}$	$69.71^{\pm 11.16}$	$56.95^{\pm 7.64}$
Ours (PageRank)	$67.40^{\pm 4.87}$	$64.14^{\pm 5.42}$	**$82.58^{\pm 1.71}$**	**$82.52^{\pm 1.73}$**	$68.95^{\pm 12.72}$	**$53.75^{\pm 6.18}$**	**$78.15^{\pm 3.47}$**	**$62.98^{\pm 8.21}$**
Ours (Betweenness)	**$80.36^{\pm 3.58}$**	**$78.24^{\pm 4.76}$**	$81.82^{\pm 1.47}$	$81.66^{\pm 1.61}$	$77.97^{\pm 2.41}$	$49.19^{\pm 1.07}$	$74.18^{\pm 10.98}$	$58.97^{\pm 9.68}$

Dataset	Energy Issues (China) [10.6%]				Insurgent Separatism (Philippines) [2.7%]			
Belief Embedding	GCN		SGVGAE		GCN		SGVGAE	
AL Method	Acc. (%)	Macro F1 (%)	Acc. (%)	Macro F1 (%)	Acc. (%)	Macro F1 (%)	Acc. (%)	Macro F1 (%)
Random	$59.84^{\pm 8.78}$	$57.44^{\pm 8.31}$	$73.67^{\pm 9.72}$	$71.47^{\pm 11.16}$	$62.16^{\pm 7.68}$	$57.36^{\pm 9.21}$	$74.96^{\pm 3.90}$	$74.77^{\pm 3.83}$
Centrality (Degree)	$58.86^{\pm 4.28}$	$58.02^{\pm 4.73}$	$75.33^{\pm 8.55}$	$74.47^{\pm 8.38}$	$47.05^{\pm 0.81}$	$46.53^{\pm 0.99}$	$73.67^{\pm 3.22}$	$73.39^{\pm 3.17}$
Centrality (PageRank)	$56.00^{\pm 3.77}$	$55.37^{\pm 4.19}$	$78.00^{\pm 7.71}$	$76.33^{\pm 9.67}$	**$66.97^{\pm 4.15}$**	$64.59^{\pm 6.09}$	$74.67^{\pm 3.98}$	$74.43^{\pm 3.94}$
Centrality (Betweenness)	$57.33^{\pm 5.59}$	$54.69^{\pm 5.33}$	$82.67^{\pm 4.13}$	$81.91^{\pm 4.20}$	$58.40^{\pm 0.75}$	$54.93^{\pm 1.07}$	$73.33^{\pm 3.55}$	$73.16^{\pm 3.53}$
Entropy	**$64.22^{\pm 8.54}$**	$57.98^{\pm 7.21}$	$75.33^{\pm 7.78}$	$74.25^{\pm 7.75}$	$59.47^{\pm 8.99}$	$55.62^{\pm 11.57}$	$73.88^{\pm 4.45}$	$73.69^{\pm 4.38}$
AGE	$52.27^{\pm 10.99}$	$47.90^{\pm 9.95}$	$78.00^{\pm 8.94}$	$76.11^{\pm 10.94}$	$56.55^{\pm 2.15}$	$52.35^{\pm 3.30}$	$73.56^{\pm 4.80}$	$73.37^{\pm 4.72}$
ANRMAB	$57.44^{\pm 9.96}$	$54.21^{\pm 9.68}$	$78.33^{\pm 9.41}$	$76.48^{\pm 11.41}$	$56.76^{\pm 4.22}$	$51.75^{\pm 3.75}$	$72.15^{\pm 5.23}$	$71.98^{\pm 5.15}$
GPA	$59.76^{\pm 6.84}$	$59.03^{\pm 6.74}$	$78.67^{\pm 8.75}$	$76.97^{\pm 10.45}$	$57.98^{\pm 2.47}$	$55.68^{\pm 3.30}$	$74.00^{\pm 5.06}$	$73.80^{\pm 4.99}$
RIM	$63.14^{\pm 5.01}$	$56.49^{\pm 3.80}$	$78.67^{\pm 5.47}$	$77.69^{\pm 5.77}$	$54.67^{\pm 0.00}$	$37.87^{\pm 0.00}$	$73.89^{\pm 3.39}$	$73.64^{\pm 3.39}$
Ours (PageRank)	$58.86^{\pm 8.55}$	$53.74^{\pm 7.85}$	$82.67^{\pm 2.31}$	$82.19^{\pm 2.36}$	$60.57^{\pm 3.34}$	$58.99^{\pm 4.07}$	$77.14^{\pm 3.12}$	$76.86^{\pm 3.06}$
Ours (Betweenness)	$60.57^{\pm 14.86}$	**$59.43^{\pm 14.81}$**	**$83.00^{\pm 3.83}$**	**$82.43^{\pm 4.14}$**	$65.33^{\pm 1.89}$	**$64.78^{\pm 1.81}$**	**$78.33^{\pm 1.28}$**	**$78.06^{\pm 1.25}$**

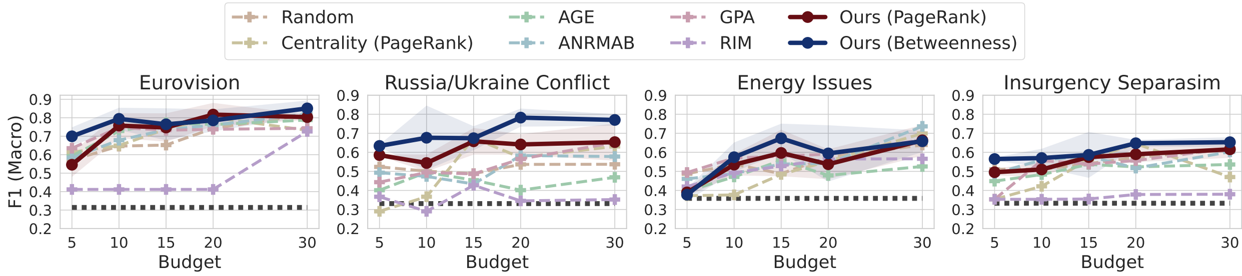
- **GPA** [14]: In this method, active learning is framed as a Reinforcement Learning (RL) problem. An agent is trained to select nodes based on the current graph state, which is implicitly modeled by a classification GNN.
- **RIM** [23]: This method uses influence functions to estimate the impact of selected samples and scales the influence-based score by label reliability. AL is formulated as a reliable influence maximization problem.

3) Metrics and Hyperparameters: We assess the quality of our queried nodes with other AL methods by evaluating the representation-based belief classification using accuracy and macro F1 score. We employ two belief representation learning models, the GCN [20] and a specialized model called SGVGAE [5] that also performs well unsupervised. In evaluating the belief classification, we focus on assertions. User preferences can be inferred by aggregating their post history, but this may compound our goal of assessing the quality of the chosen active learning nodes. Therefore, reported accuracy and macro F1 score are only for assertions.

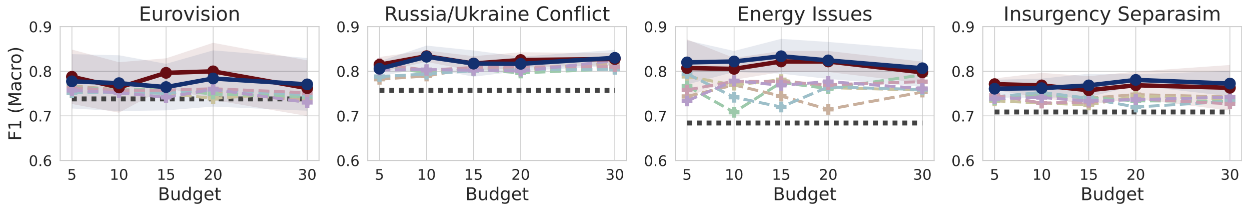
We address the budget constraint directly rather than using a percentage of the dataset size. We believe the labeling effort is proportional to the absolute number of labeling tasks and will not change with the size of the data corpus. Therefore, we notice a broad range of percentage values in our six datasets, offering another perspective for observation.

Experiments are repeated ten times with three different train-validation splits, and we report the means and standard deviations. The following are for the hyperparameters. For both variants of the PerbALGraph, which employ different centrality metrics, the default number of perturbations is 10 ($a_e = 4, a_m = 3, a_p = 3$). For 5 perturbed graphs, the configuration is (2, 2, 1), and for 15 perturbed graphs, the configuration is (6, 5, 4). The learning model within the PerbALGraph, by default, is a two-layered GCN with 32 hidden dimensions and 16 output dimensions.

We trained the GCN using the Adam optimizer with a learning rate of 0.02 and a weight decay of 0.005. This also applies to the downstream GCN encoder. When testing downstream task performance using the SGVGAE model, we initialized



(a) Belief representation learning using GCN.



(b) Belief representation learning using SGVGAE.

Fig. 2: Macro F1 score curve of belief representation models trained semi-supervised on AL-queried nodes under different budget constraints. The dotted flat gray line represents the performance of the model trained in an unsupervised fashion.

it with 32 hidden dimensions and two output dimensions. We trained the SGVGAE using the Adam optimizer with a learning rate of 0.2 and a weight decay of 0.0001.

B. Evaluation Results

1) Belief Representation Learning with A Budget of 20 Queries: The detailed evaluation results are presented in Table II. Our method has improved belief classification performance across all six datasets. Depending on the nature of the graph, either the PageRank or the betweenness variant consistently produced valuable node selection for the semi-supervised downstream task to achieve the highest performance, demonstrating its effectiveness. The average improvements in macro F1 score for both GCN and SGVGAE are 5.22% and 3.60%, respectively. Our method also consistently achieves the second-highest place.

It was also observed that when a baseline method outperforms ours, it typically utilizes graph structural statistics and heuristics (centrality and AGE). Although we also employ these metrics, we use them differently, which will be further discussed in the ablation study. Conversely, GPA and RIM methods are not leading. One possible reason is that our graph is bipartite and consists of two types of nodes. While these methods allow for masking when selecting the next query, they do not adapt as effectively as statistical and heuristic methods.

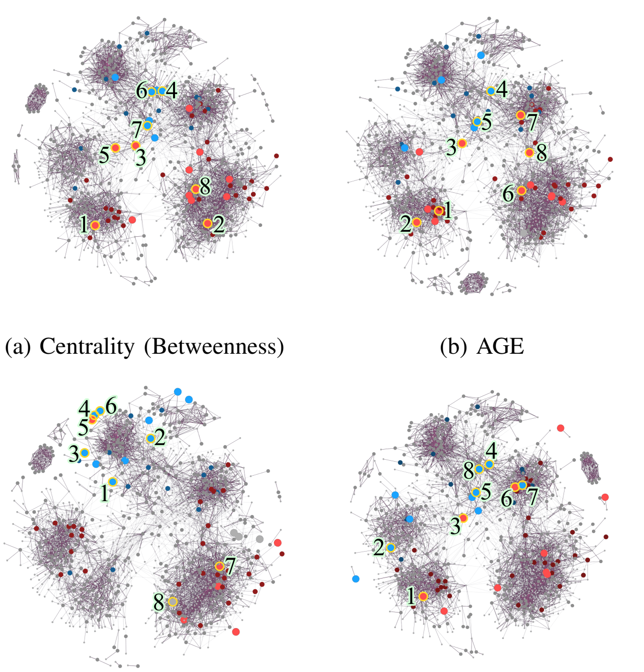
2) Belief Representation Learning with Other Allocated Budget Amounts: We also investigate the performance of our model under different budget constraints. The corresponding plots can be found in Figure 2. The dark lines in the plots represent the two variants mentioned in the table. It is evident from the plot that our method consistently outperforms other baselines. We observe that SGVGAE efficiently utilizes labeled data and maintains consistent performance, irrespective of the budget. Conversely, GCN demonstrates noticeable improvement as the budget increases.

3) Examination and Comparison of Queried Nodes: To further validate the capabilities of PerbALGraph, we visualized the queried nodes in the EDCA dataset using a budget of 20 in a graph visualization tool called Cosmograph [24]. The results can be seen in Figure 3. We have provided clear labels indicating (1) the order of the queries up to the eighth node, (2) the query labels marked in red and blue, and (3) the nodes and their labels in the test set shown in darker red and blue. We observed that our method efficiently utilizes the budget by selecting nodes 1 and 2, which effectively delineate two clusters. It also identifies node 3 as an important node to anchor. Additionally, selecting nodes 6 and 7 is also crucial because they represent conflicting yet interconnected nodes. Our method prioritizes labeling nodes at the intersection of clusters while also allowing for labeling isolated nodes, provided that the budget permits.

We also examine the queried nodes for the Russia/Ukraine Conflict (Visual) dataset in Figure 4. Our method selected important visual assertions, including aspects of politics, war, and economics. The centrality-based method gives a meaningful first query, while AGE and GPA waste the first query.

C. Ablation Study

In this study, we aim to explore how the performance of the PerbALGraph is influenced by the number of perturbed graphs. Additionally, we seek to examine the distinction between sensitivity scores and direct centrality metrics. We look into the performance of belief representation learning using SGVGAE with both variants of PerbALGraph across two datasets. By varying the number of perturbations, we analyze the range



(a) Centrality (Betweenness) (b) AGE

(c) GPA (d) PerbALGraph (PageRank)

Fig. 3: Visualization of EDCA dataset with 20 queried nodes. Bright red and blue are labeled query nodes, and the numbering indicates the selection order. Darker red and blue are test nodes with their ground-truth label.

Fig. 4: Comparison of queried visual assertions in the Russia/Ukraine Conflict dataset by different AL methods.

from direct centrality calculation ("Raw") to sensitivity score calculation with 5, 10, and 15 perturbations. Given the nature of the instability score, we cannot eliminate graph perturbation for the "Raw" variant, and we use the default value of 10 perturbations for instability score calculation.

For PageRank, the raw metric is sufficiently informative, and the PerbALGraph sees a performance dip at 5 perturbations. We hypothesize that not having enough perturbed graphs may lead to a high dynamic range in the sensitivity score, making it less meaningful than the raw PageRank metric. Conversely, the betweenness variant benefits from the sensitivity score from the start. We notice performance increases in both variants when the number of perturbations reaches 10, which is the default setting in this paper and demonstrates the effectiveness of the sensitivity score derived from the raw metrics. We also observed a drop at 15 perturbations, likely due to too many samples, causing over-smoothing and a less indicative sensitivity score.

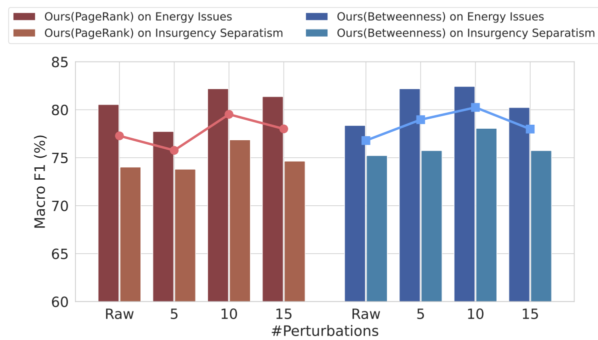


Fig. 5: The Macro F1 score of belief representation learning using SGVGAE with both variants of PerbALGraph in two datasets. The number of perturbations affects how the sensitivity score is calculated. The line plots indicate average trends.

D. Computational Costs

Adding more perturbed graphs that require GNN encoding and centrality metric computation will incur costs. However, these costs are usually one-time expenses or considered acceptable compared to the downstream tasks, which will likely take much longer to work with the selected nodes. We measure the running time of the AL process for different numbers of perturbations, and we use three datasets with increasing graph sizes to illustrate the trend when working with larger datasets. The results are presented in Figure 6, which includes the running time and a multiplier compared to the "Raw" variant. We also measured the AGE running time, which differs by only 0.5 to 1 second from the "Raw" variant. The running time for PageRank centrality increases at an acceptable rate, while the running time for betweenness centrality grows rapidly. We acknowledge this phenomenon and recommend using PageRank centrality for larger datasets. In the future, PerbALGraph can be adapted to compute a locally estimated version of betweenness centrality to address this issue.

V. Related Work

A. Semi-Supervised Graph Representation Learning

Graph representation learning [25] is a fundamental problem in graph-based machine learning, which has been applied to various tasks [26]–[28]. Unsupervised [3], [29] and semi-supervised [5], [20], [30]–[33] graph learning models have proven to be particularly effective and robust when available annotations are limited. Graph representation learning has been explored through approaches such as collaborative filtering [34], [35], non-negative matrix factorization [8], random walks [36], [37], and GCNs [38]. Recent works have

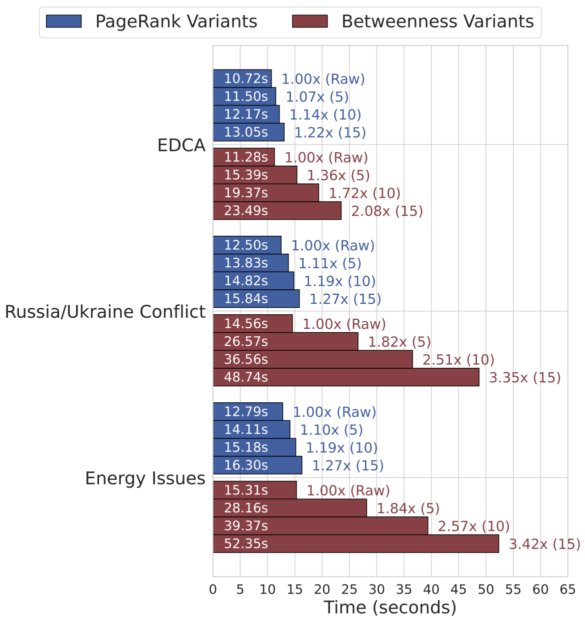


Fig. 6: Increase in total time elapsed for PerbALGraph to complete the node queries with a budget of 20 (ignoring the time for labeling the queried nodes) as the number of perturbations grows and as the size of the dataset grows.

investigated graph learning models based on graph auto-encoders [29]. For instance, InfoVGAE [3] proposes an unsupervised objective to create a disentangled non-negative latent space that preserves orthogonality and interoperability, and SGVGAE [5] proposes a semi-supervised training objective combining the disentangled embeddings with large language model-generated soft annotations to improve the performance on sparse or noisy graph data. However, most of them rely on a random strategy or simple degree-based heuristics for selecting node annotations for semi-supervision, which limits the effectiveness of node representation learning.

B. Active Learning

Active learning [9] is a machine learning paradigm that aims to reduce labeling costs by selecting the most informative examples for labeling. Recently, there has been growing interest in applying active learning to graph-structured data. In general, node selection relies on metrics that measure the informativeness of each node. Early method [10] leveraged structural properties like centrality measures (e.g., degree, PageRank [18], betweenness [17]) or assumed homophily property [39], [40] (which suggests that neighboring nodes are more likely to share the same label). Later methods have leveraged the power of GNNs to design more effective selection criteria. Uncertainty-based approaches [11], [41] prioritize nodes where the internal learning model exhibits high prediction uncertainty. For example, entropy quantifies label uncertainty, and nodes with higher entropy are selected for labeling. These metrics, however, only capture informativeness from a single type of measurement. To overcome this limitation, methods such as AGE [12] combine multiple heuristics using a weighted linear combination, with weights sampled from a time-sensitive beta distribution. ANRMAB [13] uses a multi-armed bandit approach to adjust heuristic weights dynamically. Lastly, there are other formulations of the problem that try to extract more information. GPA [14] formulates the AL process as a reinforcement learning problem with a GNN-based policy network to select nodes for long-term utility, while RIM [23] formulates the AL process as a reliable influence maximization problem.

Despite these advancements, they are not tuned to social graphs, which tend to be bipartite, heterogeneous, and have many nodes with low connectivity. GPA and RIM face this issue most acutely since their design does not accommodate bipartite social graphs. In contrast, the PerbALGraph utilizes graph perturbation and chooses nodes by assessing performance variances, leveraging more from metrics, and improving generalization across various social graphs that potentially include multi-modal nodes, such as images.

VI. Discussion and Future Work

Our method enhances active learning for belief representation on social graphs. Unlike many baselines that work less well on such bipartite graphs, PerbALGraph leverages graph perturbations to derive dynamic instability and sensitivity scores. This approach uncovers richer, more nuanced signals about a node's informativeness, especially for tasks on social networks, leading to demonstrably superior node selection for belief representation learning. Our problem formulation adopts a fixed budget rather than a percentage, reflecting practical annotation resource limitations, and we leave the question of the best budget for different graph sizes for future research. A key limitation is the increased computational cost, including an overhead of a factor $\mathbf{a}$ due to processing multiple perturbed graph instances; this is most notable with the betweenness variant, and we recommend using the PageRank variant for huge graphs. Looking ahead, promising future work includes generalization to a broader array of graph types and downstream tasks, and adapting its strategies for evolving temporal social graphs where node importance can shift.

VII. Conclusion

In this paper, we addressed the challenge of optimizing node selection for labeling in semi-supervised belief representation learning in social networks. We introduced PerbALGraph, a novel perturbation-based active learning strategy inspired by graph data augmentation, leveraging graph perturbations to select nodes based on performance variance. The proposed method integrates sensitivity and instability metrics derived from graph perturbations, effectively identifying valuable nodes for labeling. Additionally, a de-duplication module further optimizes labeling efficiency by preventing redundant annotations. Extensive experiments on six diverse social media datasets demonstrated that our method consistently outperforms existing active learning techniques, highlighting its potential for improving inference quality in social networks by better exploiting the limited labeling budget.

ACKNOWLEDGMENT

Research reported in this paper was sponsored in part by the DARPA award HR001121C0165, HR00112290105, HR00112430325 (BRIES), the Boeing Company, and the DoD Basic Research Office award HQ00342110002. It was also partially supported by ACE, one of the seven centers in JUMP 2.0, a Semiconductor Research Corporation (SRC) program sponsored by DARPA.

REFERENCES

[1] Z. Xiao, W. Song, H. Xu, Z. Ren, and Y. Sun, "TIMME: Twitter ideology-detection via multi-task multi-relational embedding," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2020, pp. 2258–2268.

[2] C. Yang, J. Li, R. Wang, S. Yao, H. Shao, D. Liu *et al.*, "Hierarchical overlapping belief estimation by structured matrix factorization," *arXiv preprint arXiv:2002.05797*, 2020.

[3] J. Li, H. Shao, D. Sun, R. Wang, Y. Yan, J. Li *et al.*, "Unsupervised belief representation learning with information-theoretic variational graph auto-encoders," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2022, pp. 1728–1738.

[4] D. Sun, C. Yang, J. Li, R. Wang, S. Yao, H. Shao *et al.*, "Computational modeling of hierarchically polarized groups by structured matrix factorization," *Frontiers in Big Data*, vol. 4, 2021.

[5] J. Li, R. Han, C. Sun, D. Sun, R. Wang, J. Zeng *et al.*, "Large language model-guided disentangled belief representation learning on polarized social graphs," in *2024 33rd International Conference on Computer Communications and Networks*, 2024, pp. 1–9.

[6] X. Liu, J. Li, D. Sun, R. Wang, T. Abdelzaher, M. Brown *et al.*, "Unsupervised image classification by ideological affiliation from user-content interaction patterns," *arXiv preprint arXiv:2305.14494*, 2023.

[7] R. Dong, Y. Sun, L. Wang, Y. Gu, and Y. Zhong, "Weakly-guided user stance prediction via joint modeling of content and social interaction," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 1249–1258.

[8] M. T. Al Amin, C. Aggarwal, S. Yao, T. Abdelzaher, and L. Kaplan, "Unveiling polarization in social networks: A matrix factorization approach," in *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, 2017, pp. 1–9.

[9] B. Settles, "Active learning literature survey," University of Wisconsin–Madison, Computer Sciences Technical Report 1648, 2009.

[10] S. P. Borgatti and M. G. Everett, "A graph-theoretic perspective on centrality," *Social Networks*, vol. 28, pp. 466–484, 2006.

[11] B. Settles and M. Craven, "An analysis of active learning strategies for sequence labeling tasks," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2008, pp. 1070–1079.

[12] H. Cai, V. W. Zheng, and K. C.-C. Chang, "Active learning for graph embedding," *arXiv preprint arXiv:1705.05085*, 2017.

[13] L. Gao, H. Yang, C. Zhou, J. Wu, S. Pan, and Y. Hu, "Active discriminative network representation learning," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 2142–2148.

[14] S. Hu, Z. Xiong, M. Qu, X. Yuan, M.-A. Côté, Z. Liu, and J. Tang, "Graph policy network for transferable active learning on graphs," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020, pp. 10 174–10 185.

[15] T. Zhao, Y. Liu, L. Neves, O. Woodford, M. Jiang, and N. Shah, "Data augmentation for graph neural networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 11 015–11 023, 2021.

[16] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, and Y. Shen, "Graph contrastive learning with augmentations," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020, pp. 5812–5823.

[17] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, vol. 40, p. 35, 1977.

[18] M. A. Rodriguez, "Grammar-based random walkers in semantic networks," *Knowledge-Based Systems*, vol. 21, pp. 727–739, 2008.

[19] Y. Zhang, M. Lease, and B. Wallace, "Active discriminative text representation learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, 2017.

[20] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[21] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal *et al.*, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning*, 2021, pp. 8748–8763.

[22] Y. Zhang, M. Jiang, Y. Meng, Y. Zhang, and J. Han, "PIEClass: Weakly-supervised text classification with prompting and noise-robust iterative ensemble training," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023, pp. 12 655–12 670.

[23] W. Zhang, Y. Wang, Z. You, M. Cao, P. Huang, J. Shan *et al.*, "RIM: reliable influence-based active learning on graphs," in *Proceedings of the 35th International Conference on Neural Information Processing Systems*, 2021.

[24] N. Rokotyan, O. Stukova, D. Kolmakova, and D. Ovsyannikov, "Cosmograph: GPU-accelerated force graph layout and rendering." [Online]. Available: https://cosmograph.app/

[25] F. Chen, Y.-C. Wang, B. Wang, and C.-C. J. Kuo, "Graph representation learning: a survey," *APSIPA Transactions on Signal and Information Processing*, vol. 9, 2020.

[26] J. Bruna and X. Li, "Community detection with graph neural networks," *stat*, vol. 1050, p. 27, 2017.

[27] J. Jiang, X. Ren, and E. Ferrara, "Social media polarization and echo chambers in the context of COVID-19: Case study," *arXiv preprint arXiv:2103.10979*, 2021.

[28] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, pp. 75–174, 2010.

[29] T. N. Kipf and M. Welling, "Variational graph auto-encoders," *arXiv preprint arXiv:1611.07308*, 2016.

[30] Z. Song, X. Yang, Z. Xu, and I. King, "Graph-based semi-supervised learning: A comprehensive review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, pp. 8174–8194, 2023.

[31] R. Wang, B. Li, Y. Lu, D. Sun, J. Li, Y. Yan *et al.*, "Noisy positive-unlabeled learning with self-training for speculative knowledge graph reasoning," in *Findings of the Association for Computational Linguistics: ACL 2023*, 2023, pp. 2440–2457.

[32] R. Wang, Z. Li, D. Sun, S. Liu, J. Li, B. Yin, and T. Abdelzaher, "Learning to sample and aggregate: few-shot reasoning over temporal knowledge graphs," in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, 2022, pp. 16 863–16 876.

[33] R. Wang, Y. Zhang, J. Li, S. Liu, D. Sun, T. Wang, T. Wang, Y. Chen, D. Kara, and T. Abdelzaher, "Metahkg: Meta hyperbolic learning for few-shot temporal reasoning," in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024, pp. 59–69.

[34] X. Yang, Y. Guo, Y. Liu, and H. Steck, "A survey of collaborative filtering based social recommender systems," *Computer Communications*, vol. 41, pp. 1–10, 2014.

[35] M. C. Pham, Y. Cao, R. Klamma, and M. Jarke, "A clustering approach for collaborative filtering recommendation using social network analysis," *JUCS - Journal of Universal Computer Science*, vol. 17, pp. 583–604, 2011.

[36] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: online learning of social representations," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 701–710.

[37] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: Large-scale information network embedding," in *Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 1067–1077.

[38] Q. Li, Z. Han, and X.-m. Wu, "Deeper insights into graph convolutional networks for semi-supervised learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[39] A. Gadde, A. Anis, and A. Ortega, "Active semi-supervised learning using sampling theory for graph signals," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 492–501.

[40] Q. Gu, C. Aggarwal, J. Liu, and J. Han, "Selective sampling on graphs for classification," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2013, pp. 131–139.

[41] M. Bilgic, L. Mihalkova, and L. Getoor, "Active learning for networked data," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 2010, pp. 79–86.