1 OUTLINE OF SUPPLEMENTARY MATERIAL

This supplemental material is organized as follows:

Section 2 Proof of reliability calculation.
Section 3 Dataset description.
Section 4 Implementation details.

2 PROOF OF RELIABILITY CALCULATION

2.1 Problem Description and Assumptions

In the following, when talking about reliability of a label, we refer to the probability that the label is correct.

Formally, for every sample we choose to be labeled by the oracle, we will eventually obtain the following related labels:

- One label from the model, corresponding to that sample.
- One or two labels from the oracle(s), the first corresponding to that sample, and the second corresponding to a similar sample, if existing (in this case, the second sample will be labeled by a different oracle to ensure independence).

The known conditions are:

- For a given sample, the accuracy of the model is denoted by β, which is equal to the accuracy on the validation set times the confidence of the model. Therefore, β changes w.r.t. the sample and the time (with training proceeding, the model is expected to become more reliable).
- There are k classes in total.
- For every sample we choose, the accuracy of the oracle who labels this sample is denoted by α_1. If a similar sample is also chosen, the accuracy of the oracle who labels this similar sample is denoted by α_2. For simplicity, we just refer to these two oracles as the first and the second oracle, and the corresponding accuracy is α_1, α_2 respectively. Note that the second oracle does not necessarily exist.

We have made the following assumptions.

- For each sample s and an oracle with accuracy α, the probability that the oracle gives a correct label only depends on α. That being said, given an oracle, the probability of correctness is a constant.
- Labels given by different oracles are independent, even if different oracles may have different accuracy, i.e., their proficiency may differ.
- Labels given by the oracle and the model are independent.
- In this supplementary material, when we say two samples are similar, we expect that they share the same label.
- When a sample is mislabeled by either the oracle or the model, the wrong label will be chosen uniformly at random from the remaining $k-1$ classes. That being said, if the accuracy is γ, then a certain wrong label will be given with the probability of $\frac{1-\gamma}{1-k}$.

According to our assumptions, we have two or three labels given by either the oracle or the model, and the corresponding ground truth label (the real label of the sample) should be the same. What's more, the reliability of these labels are independent.

Our goal is:

- Given α, β, k, and labels (given by the oracle and the model, may be two or three in total), calculate the reliability of our final decision. Specifically, if there are just two labels, or there are three labels and two of them are the same, we will pick one of them and we will want to know the reliability of our choice.

2.2 Framework

We propose a two-stage solution to this problem, and these two stages are fully decoupled:

- Stage 1 (Theorem 4.1): If there are two labels given by oracles (the reliability of them are α_1 and α_2, respectively), since we always pick the latter, we should calculate the reliability of it and regard it as the combined reliability of the oracles (denoted by α'). If there is just one, just set the combined reliability as α_1. After Stage 1, we have one combined label given by the oracles, and one label given by the model whose reliability is β.
- Stage 2 (Theorem 4.2): Now we have exactly one label from the oracles with the reliability α' and one from the model with the reliability β. We can thus calculate the final reliability of the label we pick, if applicable (in some cases, we just discard all labels).

2.3 Details of Stage 1 (Theorem 4.1)

We just deal with the non-trivial case here, namely the situation where we have two labels from the oracle. According to whether these two labels are identical, we divide the problem into two cases.

2.3.1 Case 1: Two oracles give the same label.

Theorem 1. *If two oracles give the same label, the reliability of these two labels are independent and the reliability of them is α_1 and α_2 respectively, then the reliability of the latter (i.e., the combined reliability) is*

$$\frac{\alpha_1\alpha_2}{\alpha_1\alpha_2 + \frac{(1-\alpha_1)(1-\alpha_2)}{k-1}} \tag{1}$$

In this case, these two labels can be both correct or incorrect. The probability that both are correct is

$$P(both\ are\ correct) = \alpha_1 * \alpha_2 = \alpha_1\alpha_2 \tag{2}$$

The correctness of the second sign is supported by the independence of these two trials. The probability that both are incorrect is

$$P(both\ are\ incorrect) = (k-1) * \frac{1-\alpha_1}{k-1} * \frac{1-\alpha_2}{k-1} = \frac{(1-\alpha_1)(1-\alpha_2)}{k-1} \tag{3}$$

The reason for this equation is that: for the first oracle, there are $k-1$ wrong cases in total, and the probability of each is $\frac{1-\alpha_1}{k-1}$ due to the uniform assumption. This also applies to the second oracle, except that α_1 should be replaced by α_2.

Thus, the reliability of the latter label (namely the combined reliability of labels given by the oracles) is given by

$$P(both\ are\ correct \mid the\ oracle\ gives\ the\ same\ label\ twice) = \frac{P(both\ are\ correct)}{P(both\ are\ correct) + P(both\ are\ incorrect)} = \frac{\alpha_1\alpha_2}{\alpha_1\alpha_2 + \frac{(1-\alpha_1)(1-\alpha_2)}{k-1}} \tag{4}$$

2.3.2 Case 2: Two oracles give different labels.

Theorem 2. *If two oracle give two different labels, the reliability of these two labels are independent and the reliability of them is α_1 and α_2 respectively, then the reliability of the latter (i.e., the combined reliability) is*

$$\frac{(1-\alpha_1)\alpha_2}{1-\alpha_1\alpha_2}. \tag{5}$$

In this case, the oracle can only be correct at most once. The probability that the oracle is correct at the second time (thus incorrect at the first time) is given by

$$P(the\ first\ time\ is\ incorrect,\ the\ second\ time\ is\ correct) = (1-\alpha_1) * \alpha_2 = (1-\alpha_1)\alpha_2. \tag{6}$$

The correctness of this formula is, again, due to the independence of these two trials. The probability that two oracle give two different labels is given by

$$P(the\ oracle\ gives\ two\ different\ labels) = 1 - P(both\ are\ correct) = 1 - \alpha_1\alpha_2, \tag{7}$$

where $P(both\ are\ correct)$ refers to the same thing as that in Case 1 – the oracle is correct twice. Therefore, the reliability of the latter label can be calculated as follows

$$P(the\ first\ time\ is\ incorrect,\ the\ second\ time\ is\ correct \mid the\ oracle\ gives\ two\ different\ labels) = \frac{(1-\alpha_1)\alpha_2}{1-\alpha_1\alpha_2} \tag{8}$$

2.3.3 Brief summary.

To summarize, the combined reliability of the oracle is given by

$$\alpha' = \begin{cases} \alpha_1 & the\ oracle\ gives\ one\ label \\ \frac{\alpha_1\alpha_2}{\alpha_1\alpha_2 + \frac{(1-\alpha_1)(1-\alpha_2)}{k-1}} & two\ oracles\ give\ the\ same\ label \\ \frac{(1-\alpha_1)\alpha_2}{1-\alpha_1\alpha_2} & two\ oracles\ give\ different\ labels \end{cases} \tag{9}$$

2.4 Details of Stage 2 (Theorem 4.2)

Now we have the combined reliability of the oracle(s) α' (note that we always take the latter label as the label given by the oracle(s)), and the reliability of the model β. And we are approaching the final result – the combined reliability of the label we pick.

This stage is indeed very similar to the previous one.

As a special note, in the following, when we say "the oracle and the model agree with each other", we mean the label we pick from the one or two labels given by the oracle is the same as the one given by the model. Naturally, when we say they disagree with each other, we mean the two aforementioned labels differ.

Once again, we divide the problems into two cases according to whether they agree with each other.

2.4.1 *The oracle and the model agree with each other.*

Theorem 3. *If the oracle and the model agree with each other, the reliability of these two labels are independent and are α' and β respectively, then the combined reliability is*

$$\alpha' * \beta = \alpha'\beta \tag{10}$$

In this case, they can only be both correct or both incorrect.

The probability that they are both correct is given by

$$P(both\ the\ oracle\ and\ the\ model\ are\ correct) = \alpha' * \beta = \alpha'\beta \tag{11}$$

The probability that they are both incorrect is given by

$$P(both\ the\ oracle\ and\ the\ model\ are\ incorrect) = (k-1) * \frac{1-\alpha'}{k-1} * \frac{1-\beta}{k-1} = \frac{(1-\alpha')(1-\beta)}{k-1} \tag{12}$$

The reason for the above two formulas is very similar to the that in the first case of Stage 1, that is, due to the independence and the uniform assumption.

Thus, the reliability of the combined reliability is given by

$$\begin{aligned} &P(both\ the\ oracle\ and\ the\ model\ are\ correct \mid the\ oracle\ and\ the\ model\ agree\ with\ each\ other) \\ &= \frac{P(both\ the\ oracle\ and\ the\ model\ are\ correct)}{P(both\ the\ oracle\ and\ the\ model\ are\ correct) + P(both\ the\ oracle\ and\ the\ model\ are\ incorrect)} = \frac{\alpha'\beta}{\alpha'\beta + \frac{(1-\alpha')(1-\beta)}{k-1}} \end{aligned} \tag{13}$$

2.4.2 *The oracle and the model disagree with each other.*

Theorem 4. *If the oracle and the model disagree with each other, the reliability of these two labels are independent and are α' and β respectively, then the reliability of the oracle is*

$$\frac{\alpha'(1-\beta)}{1-\alpha'\beta} \tag{14}$$

For simplicity, we just let the label we pick from the oracle represent the opinion of the oracle. Thus, when we say the oracle is correct in this case, we mean the label we pick from the oracle is correct in reality, and vice versa.

In this case, the oracle and the model can be either correct or both incorrect.

The probability that the oracle is correct (thus the model being incorrect) is given by

$$P(the\ oracle\ is\ correct\ ,\ the\ model\ is\ incorrect) = \alpha' * (1-\beta) = \alpha'(1-\beta) \tag{15}$$

The probability that the model is correct (thus the oracle being incorrect) is given by

$$P(the\ oracle\ is\ incorrect\ ,\ the\ model\ is\ correct) = (1-\alpha') * \beta = (1-\alpha')\beta \tag{16}$$

The independence of the oracle and the model is utilized again to show the correctness of the above two formulas.

The probability that the oracle and the model disagree with each other is given by

$$P(the\ oracle\ and\ the\ model\ disagree\ with\ each\ other) = 1 - P(both\ the\ oracle\ and\ the\ model\ are\ correct) = 1 - \alpha'\beta \tag{17}$$

note that $P(both\ the\ oracle\ and\ the\ model\ are\ correct)$ in the previous case in Stage 2 refers to exactly the same thing.

Therefore, the reliability of the oracle (indeed the label we pick from the oracle) and the model in this case can be calculated as follows respectively:

$$\begin{aligned} &P(the\ oracle\ is\ correct,\ the\ model\ is\ incorrect \mid the\ oracle\ and\ the\ model\ disagree\ with\ each\ other) \\ &= \frac{P(the\ oracle\ is\ correct\ ,\ the\ model\ is\ incorrect)}{P(the\ oracle\ and\ the\ model\ disagree\ with\ each\ other)} = \frac{\alpha'(1-\beta)}{1-\alpha'\beta} \end{aligned} \tag{18}$$

with the former corresponding to the reliability of the oracle, and the latter corresponding to the reliability of the model.

2.4.3 *Brief summary.*

To summarize, the reliability of the oracle is given by

$$\begin{cases} \frac{\alpha'\beta}{\alpha'\beta + \frac{(1-\alpha')(1-\beta)}{k-1}} & the\ oracle\ and\ the\ model\ agree\ with\ each\ other \\ \frac{\alpha'(1-\beta)}{1-\alpha'\beta} & the\ oracle\ and\ the\ model\ agree\ with\ each\ other \end{cases} \tag{19}$$

and the reliability of the model is given by

$$\begin{cases} \frac{\alpha'\beta}{\alpha'\beta + \frac{(1-\alpha')(1-\beta)}{k-1}} & the\ oracle\ and\ the\ model\ agree\ with\ each\ other \\ unused & the\ oracle\ and\ the\ model\ agree\ with\ each\ other \end{cases} \tag{20}$$

Table 1: Overview of the Four Datasets

Dataset	#Nodes	#Features	#Edges	#Classes	#Train/Val/Test	Task type	Description
Cora	2,708	1,433	5,429	7	1,208/500/1,000	Transductive	citation network
Citeseer	3,327	3,703	4,732	6	1,827/500/1,000	Transductive	citation network
Pubmed	19,717	500	44,338	3	18,217/500/1,000	Transductive	citation network
ogbn-arxiv	169,343	128	1,166,243	40	90,941/29,799/48,603	Transductive	citation network
Reddit	232,965	602	11,606,919	41	155,310/23,297/54,358	Inductive	social network

2.5 Proof of Theorem 5.1

THEOREM 5. *The greedily selected batch node set is within a factor of* $(1-\frac{1}{e})$ *of the optimal set for the objective of effective influence maximization (EIM).*

Consider a batch selection setting with $\mathcal{B}/b$ rounds where b nodes are selected in each iteration (see Alg.1) and $\mathcal{B}$ denotes the labeling budget. Formally, EIM objective is equal to:

$$\max_{\mathcal{V}_b} F(\mathcal{V}_b) = |\sigma(\mathcal{V}_l \cup \mathcal{V}_b)|, \mathbf{s.t.} \mathcal{V}_b \subseteq \mathcal{V} \setminus \mathcal{V}_l, |\mathcal{V}_b| = b \tag{21}$$

and σ is defined as:

$$\sigma(\mathcal{V}_l) = \bigcup_{v \in \mathcal{V}, Q(v,\mathcal{V}_l,k)>\theta} \{v\}. \tag{22}$$

where Q is defined in Definition 4.4, θ is the hyper-parameter defined in eq (12), $\mathcal{V}_l$ is the set of nodes selected in all previous rounds.

The submodularity of this objective is proved as follows.

For every $A \subseteq B \subseteq S$ and $s \in S \setminus B$, let $Q_A(v) = \max_{v_i \in \mathcal{V}_l \cup A} Q(v, v_i, k)$ and $Q_B(v) = \max_{v_j \in \mathcal{V}_l \cup B} Q(v, v_j, k)$. Since $(\mathcal{V}_l \cup A) \subseteq (\mathcal{V}_l \cup B)$, for any $v \in \mathcal{V}$, $Q_A(v) \leq Q_B(v)$. Thus we have:

$$F(A \cup s) - F(A) = |Q(v, s, k) > \theta \geq Q_A(v)| \geq |Q(v, s, k) > \theta \geq Q_B(v)| = F(B \cup s) - F(B) \tag{23}$$

Therefore, the Theorem follows.

3 DATASET DESCRIPTION

Cora, **Citeseer**, and **Pubmed**[1] are three popular citation network datasets, and we follow the public training/validation/test split in GCN [5]. In these three networks, papers from different topics are considered as nodes, and the edges are citations among the papers. The node attributes are binary word vectors, and class labels are the topics papers belong to.

Reddit is a social network dataset derived from the community structure of numerous Reddit posts. It is a well-known inductive training dataset, and the training/validation/test split in our experiment is the same as that in GraphSAGE [3]. The public version provided by GraphSAINT[2] [7] is used in our paper.

ogbn-arxiv is a directed graph, representing the citation network among all Computer Science (CS) arXiv papers indexed by MAG. The training/validation/test split in our experiment is the same as the public version. The public version provided by OGB[3]is used in our paper.

For more specifications about the five aforementioned datasets, see Table 1.

4 IMPLEMENTATION DETAILS

The threshold θ is chosen as 0.05 for two small datasets: Cora and Citesser, and 0.005 for other datasets. The γ is chosen as 0.5 for Cora, PubMed and Reddit, and 0.4 for other datasets. In terms of GPA [4], in order to obtain its full accuracy, the pre-trained model released by its authors on Github is adopted. More precisely, for Cora, we choose the model pre-trained on PubMed and Citeseer; for PubMed, we choose the model pre-trained on Cora and Citeseer; for Citeseer and Reddit, we choose the model pre-trained on Cora and PubMed. Other hyper-parameters are all consistent with the released code. When it comes to AGE [1] and ANRMAB [2], in order to obtain well-trained models and guarantee that the model-based selection criteria employed by them run well, GCN is trained for 200 epochs in each node selection iteration. AGE is implemented with its open-source version and ANRMAB in accordance with its original paper. For ALG [8] and GRAIN [9], we follow the public code released by the original paper. In addition, k (i.e., the number of classes) labels can be given by oracle in each iteration. As an instance, k is chosen as 7 in Cora.

The experiments are conducted on an Ubuntu 16.04 system with Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20GHz, 4 NVIDIA GeForce GTX 1080 Ti GPUs and 256 GB DRAM. All the experiments are implemented in Python 3.6 with Pytorch 1.7.1 [6] on CUDA 10.1.

[1]https://github.com/tkipf/gcn/tree/master/gcn/data
[2]https://github.com/GraphSAINT/GraphSAINT
[3]https://ogb.stanford.edu/docs/nodeprop/#ogbn-arxiv

REFERENCES

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

[2] L. Gao, H. Yang, C. Zhou, J. Wu, S. Pan, and Y. Hu. Active discriminative network representation learning. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 2142–2148, 2018.

[3] W. L. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1024–1034, 2017.

[4] S. Hu, Z. Xiong, M. Qu, X. Yuan, M. Côté, Z. Liu, and J. Tang. Graph policy network for transferable active learning on graphs. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[5] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[6] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019.

[7] H. Zeng, H. Zhou, A. Srivastava, R. Kannan, and V. K. Prasanna. Graphsaint: Graph sampling based inductive learning method. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

[8] W. Zhang, Y. Shen, Y. Li, L. Chen, Z. Yang, and B. Cui. ALG: fast and accurate active learning framework for graph convolutional networks. In *SIGMOD '21: International Conference on Management of Data, Virtual Event, China, June 20-25, 2021*, pages 2366–2374, 2021.

[9] W. Zhang, Z. Yang, Y. Wang, Y. Shen, Y. Li, L. Wang, and B. Cui. Grain: Improving data efficiency of graph neural networks via diversified influence maximization. *Proc. VLDB Endow.*, 14(11):2473–2482, 2021.