LLMs with Graphs

Shai Pranesh, Vedang Bhupesh Shenvi Nadkarni, Shivanshi, Varun L

Abstract

This report delves into the potential of Large Language Models (LLMs) and their crucial role in graph machine learning, particularly for node classification tasks on text data, inspired by the work of "Exploring the Potential of Large Language Models (LLMs) in Learning on Graphs". LLMs, with their proven efficacy in NLP tasks and beyond, offer rich node representations and complex reasoning capabilities, making them promising for enhancing GNNs. We focus on two LLM-based approaches: LLM-as-enhancer and LLM-as-predictor, exploring their capabilities and limitations in classifying movie reviews from the MR dataset.

1 Introduction

Graph-based learning has gained significant attention for its practical applications. The standard approach involves using Graph Neural Networks (GNNs) with shallow text embeddings for node attributes. However, this has limitations in terms of general knowledge and deep semantic understanding. Large Language Models (LLMs) have recently shown great promise in possessing extensive common knowledge and robust semantic comprehension for handling text data.

In this project, we explore the potential of LLMs in graph machine learning, specifically focusing on the node classification task. We examine two possible approaches: LLMs-as-Enhancers and LLMs-as-Predictors. The former uses LLMs to enhance nodes' text attributes with their vast knowledge and generates predictions through GNNs. The latter directly employs LLMs as standalone predictors.

Through comprehensive studies in various settings, we present our observations and insights.

2 Background

In our study, we differentiate between PLMs(pre-trained language models) and LLMs(Large Language Models) in terms of-

2.1 Size and Capability

- PLMs: Relatively smaller models with fewer parameters (e.g., BERT, Deberta). Excel at basic language tasks like word prediction and sentence completion.
- LLMs: Significantly larger models with orders of magnitude more parameters (e.g., ChatGPT, GPT-4). Capable of more complex tasks like creative text generation, translation, and reasoning.

2.2 Fine-tuning and Customization

- PLMs: Easy to fine-tune for specific downstream tasks using their pre-trained knowledge. Enables adaptation to specific domains and goals.
- LLMs: Fine-tuning can be challenging due to size and complexity. Often used as-is in closed-source services, limiting customization.

2.3 Transparency and Accessibility

- PLMs: Open-source models with accessible parameters and embeddings. Allows for research, development, and interpretability.
- LLMs: Can be open-source (e.g., LLaMA) or closed-source (e.g., ChatGPT). Closed-source models restrict access, limiting transparency and user control.

2.4 Deployment and Use Cases

- PLMs: Often deployed locally or embedded in applications. Suitable for specific tasks requiring fine-tuning and customization. Good for research and development.
- LLMs: Often deployed as cloud-based services. Ideal for general-purpose language tasks requiring advanced capabilities. Used in commercial applications and services.

3 Methodology

This section outlines our methodology for leveraging Large Language Models (LLMs) in enhancing node attributes and predicting node properties on graph data. We explore two distinct approaches:

3.1 LLM as Enhancer

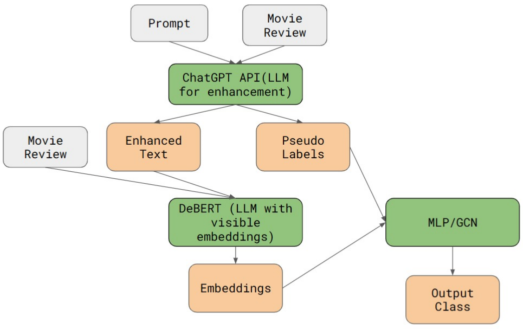


Figure 1: Model flow diagram

3.1.1 Model and its details

- We enhance the text provided with the use of LLMs like ChatGPT and other large models in which the embeddings may not be visible but have good enhancing capabilities. The enhancement is done by giving a suitable prompt and along with it pseudo-labels are obtained, which in our case was not always found to be perfectly classifying the given text. Later, it is discussed in the experiments section.

- Tokenization: We have to tokenize the enhanced text and the original text, then concatenate their embedding before passing them on to the MLP/GCN model as shown in Figure 1. For this task, any PLM or embedding visible LLM can be utilized and should be fine-tuned before use.

- Finetuning of LLM and MLP/GCNs: For the finetuning of LLM for tokenization, we used the pseudo labels to make the LLM/PLM classify and fine-tune them. By using the pseudo labels, we are not giving the actual label information into the model which the MLP/GCN has to predict. The reason to use MLP/GCN model is that the pseudo-labels are wrong in many of the case and to further improve them, we use MLP/GCN.

- Graph construction: The graph construction can be done using the similarity between the vectors obtained from the fine-tuned PLM. In the case of the MLP, the model does not need the Graph but will contain the node features in terms of vectors, which can propagate the graph information.

- In our experiments, we used ChatGPT as the LLM and DeBERT as the PLM, and we used only MLP.

3.1.2 Evaluation Metrics

- Task-specific Metrics: We evaluate the performance of the LLM-based prediction approach using task-specific metrics relevant to the problem.

3.2 LLM as Predictor

3.2.1 Prompt Design

We use the three approaches as proposed in (Chen et al., 2023). These approaches incorporate different amounts of information into the target prompt for prediction, and help us understand the underlying capacity of LLMs.

- Zero-shot queries: We use labelled data to generate a prompt which asks the LLM to predict the class of a Movie review based on the only the text of that moview review. No other information is given.

- Few-shot queries: We select a target review and then sample other reviews. We provide the LLM with information (in the prompt) about the labels of the other reviews, and then ask it to use those labels as a prior in predicting the label for the target movie review.

- Neighbour summary with query: We first run a Text-GCN(Yao et al., 2018) to create a graph which relates different reviews (as nodes) with other reviews. Then we select the movie reviews most closely related to our target review. We first generate a prompt to summarize the neighbours, and then provide the summary along with the target review as a prompt for prediction to the LLM. This is a way to incorporate graph relations implicitly into the prompt. The similarity in the neighbours captured is expected to give a better hint to the LLM for prediction.

3.2.2 LLM Inference

- Inference Mode: We choose an appropriate inference mode for the LLM. This could involve single-hop inference where the LLM directly predicts based on the provided prompt and node features, or multi-hop inference where the LLM iteratively interacts with the graph and refines its predictions.

- Post-processing (Optional): Depending on the task and LLM output, we may apply post-processing steps to refine the predictions. This could involve thresholding, normalization, or combining the LLM's outputs with other information from the graph.

3.2.3 Evaluation Metrics

- Task-specific Metrics: We evaluate the performance of the LLM-based prediction approach using task-specific metrics relevant to the problem.

4 Experiments

4.1 Dataset

MR (Movie review) dataset is a widely used benchmark dataset for sentiment analysis research. It consists of sub-datasets focusing on different aspects of movie reviews. We have used its Polarity sub-dataset- to do Binary classification of reviews as positive or negative sentiment. Its size is 2,000 reviews (1,000 positive, 1,000 negative). It contains single line of text per review and is extracted from Rotten Tomatoes and IMDB. For the performance a simple macro accuracy was used to report the robustness of the model.

4.2 Training and validation Hyper parameters

4.2.1 LLM as enhancers

The objective of the task was to classify the MR dataset according to its overall sentiments - positive (1) or negative (0) using LLM as an enhancer. The movie review dataset was given as input to LLM (ChatGPT API) with the following prompt as :

"Question: What is the review type of the text provided as Movie Review (positive or negative? Give explanation for your reasoning. Give your answer in the format in which the first line is either 1 (positive) or 0 (negative), then with 2 lines spaces in b/w give your reasoning"

The answer for this prompt contains the integer 0 or 1 for the classes, acting as the pseudo labels, and an explanation that acts as the enhanced text attribute to further computations.

The flow for the entire model is given in the Figure 1. The input for the MLP was the concatenated output of the enhanced text embeddings with the pseudo labels.

For getting reliable embeddings from DeBERT, we fine-tuned it using the pseudo labels that and making the DeBERT predict it.

AdamW with a learning rate of 1e-3 and Cross Entropy loss was used for finetuning DeBERT, and pseudo labels of size 8 was concatenated with embeddings of size 768*2 (enhanced text + input text embeddings concatenated).

4.2.2 LLM as predictors

We used GCN for neighbor finding and the results we got for the prompts used for LLM-as-Predictor are

1. **Zero-shot prompt:**
   Movie review: a culture-clash comedy that , in addition to being very funny , captures some of the discomfort and embarrassment of being a bumbling american in europe .
   Task: Classify this review into one of two categories positive or negative. Respond with 0 for negative and 1 for positive. Do not ask more questions.

   The ground truth for this is *positive*.

2. **Few-shot prompt:**
   Movie review 1: it's [ricci's] best work yet , this girl-woman who sincerely believes she can thwart the world's misery with blind good will .
   Movie review 2: nettelbeck has crafted an engaging fantasy of flavours and emotions , one part romance novel , one part recipe book .
   Movie review 3: thoroughly enjoyable .
   Review 1 is classified as positive, Review 2 is classified as positive, Review 3 is classified as positive. Task: Classify the following review review into one of two categories positive or negative. Respond with 0 for negative and 1 for positive. Do not ask more questions.
   Movie review:it's a drawling , slobbering , lovable run-on sentence of a film , a southern gothic with the emotional arc of its raw blues soundtrack .

   The ground truth for the target review is *positive*, and here 3 labelled reviews are provided.

The choice is rather arbitrary for our experiments.

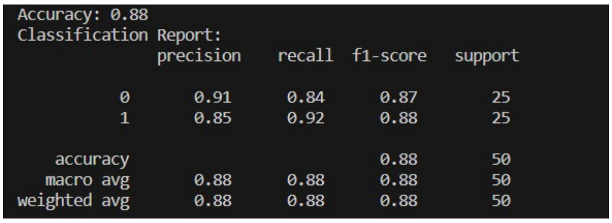
3. **Neighbour summarization:**

   - **Neighbour summarization response from chatgpt:**
     The three movie reviews share a common thread of praising distinct aspects of the films they discuss. Review 1 applauds a film centered around a family's joyful life on the Yiddish stage, highlighting the theme of celebration. Review 2 commends director Kapur for his adeptness with epic landscapes and adventure, positioning the film as an improvement over his earlier work. Lastly, Review 3 draws a parallel between the original Japanese film "Ringu" and its Americanized adaptation, drawing a comparison akin to the evolution from "Evil Dead" to "Evil Dead II." Despite addressing different genres and cultural contexts, the commonality lies in the positive evaluations of specific elements, such as family celebration, directorial skill, and successful adaptation.
   - **Prediction prompt:**
     Movie review: pray's film works well and will appeal even to those who aren't too familiar with turntablism .
     Neighbour summary: The three movie reviews share a common thread of praising distinct aspects of the films they discuss. Review 1 applauds a film centered around a family's joyful life on the Yiddish stage, highlighting the theme of celebration. Review 2 commends director Kapur for his adeptness with epic landscapes and adventure, positioning the film as an improvement over his earlier work. Lastly, Review 3 draws a parallel between the original Japanese film "Ringu" and its Americanized adaptation, drawing a comparison akin to the evolution from "Evil Dead" to "Evil Dead II." Despite addressing different genres and cultural contexts, the commonality lies in the positive evaluations of specific elements, such as family celebration, directorial skill, and successful adaptation.
     Task: Classify this review into one of two categories positive or negative. Respond with 0 for negative and 1 for positive. Do not ask more questions. The ground truth for the review is *negative*.

4.3 Overall Performance

We tested it with a limited dataset size of 180 with 50 movie reviews as validation and the rest for training. We could not test on the actual whole dataset as the ChatGPT API had only 200 requests per day as a free tier.

The results on this limited dataset gave about 0.88 percent accuracy with 10 iterations of finetuning on the DeBERT model for LLM as enhancers. Detailed results are given in Figure 2.



```
Accuracy: 0.88
Classification Report:
              precision    recall  f1-score   support

           0       0.91      0.84      0.87        25
           1       0.85      0.92      0.88        25

    accuracy                           0.88        50
   macro avg       0.88      0.88      0.88        50
weighted avg       0.88      0.88      0.88        50
```

Figure 2: Results for LLM as enhancer

4.4 Potential Limitations

- **Computational Costs of LLMs** Querying LLMs for text augmentation can be expensive, especially for large datasets. Exploring more efficient LLM utilization methods or alternative text augmentation techniques that require fewer LLM calls could be beneficial.

- **Domain Specificity** The MR dataset focuses on movie reviews, limiting the generalizability of the findings. Testing the methods on other sentiment analysis domains (e.g., product reviews, and social media posts) would assess their broader applicability.

- **Limited Dataset Size** While the MR dataset provides valuable insights, its relatively small size may not fully capture the complexities of real-world sentiment analysis tasks. Expanding the experiment to larger and more diverse datasets could offer a more robust evaluation of the proposed methods.

4.5 Future Work

We further plan to test the entire MR dataset by getting the paid ChatGPT API tier and comparing them extensively with other models like TextGCN. Also, we aim to use different models like GCN and GAT instead of MLP for the last part of classification as stated in the original paper.

Contributions: LLM as Enhancers(Shai and Varun), LLM as Predictors(Shivanshi and Vedang)

The code is available at this link : https://github.com/shaipranesh2/Graph-mining-assignment/tree/main

References

Zhikai Chen, Haitao Mao, Hang Li, Wei Jin, Hongzhi Wen, Xiaochi Wei, Shuaiqiang Wang, Dawei Yin, Wenqi Fan, Hui Liu, and Jiliang Tang. 2023. Exploring the potential of large language models (llms) in learning on graphs.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2018. Graph convolutional networks for text classification.