Which Modality should I use - Text, Motif, or Image? : Understanding Graphs with Large Language Models

Anonymous ACL submission

Abstract

Our research integrates graph data with Large Language Models (LLMs), which, despite their advancements in various fields using large text corpora, face limitations in encoding entire graphs due to context size constraints. This paper introduces a new approach to encoding a graph with diverse modalities, such as text, image, and motif, coupled with prompts to approximate a graph's global connectivity, thereby enhancing LLMs' efficiency in processing complex graph structures. The study also presents GRAPHTMI, a novel benchmark for evaluating LLMs in graph structure analysis, focusing on homophily, motif presence, and graph difficulty. Key findings indicate that the image modality, especially with vision-language models like GPT-4V, is superior to text in balancing token limits and preserving essential information and outperforms prior graph neural net (GNN) encoders. Furthermore, the research assesses how various factors affect the performance of each encoding modality and outlines the existing challenges and potential future developments for LLMs in graph understanding and reasoning tasks. All data will be publicly available upon acceptance.

1 Introduction

Large Language Models (LLMs) are increasingly utilized in areas with inherent graph structures like social network analysis (Mislove et al., 2007), drug discovery (Vishveshwara et al., 2002), and recommendation systems (Melville and Sindhwani, 2010), but they face limitations due to their reliance on unstructured text and challenges in incorporating new data post-training (Zhang et al., 2023; Lewis et al., 2020; Pan et al., 2023). Graph-structured data can address these issues, providing a nuanced and flexible representation of real-world relationships.

While there has been progress in interpreting multi-modal information (Yin et al., 2023), integrating graph understanding into LLMs remains a developing area. Current challenges include LLMs' difficulty directly processing complex graph-structured data, necessitating innovative input encoding and prompt design (Fatemi et al., 2023; Chen et al., 2023) for various graph tasks. Current research typically employs limited setups with small real-world graphs (Guo et al., 2023) or synthetic ones (Wang et al., 2023), exposing a gap in effectively incorporating large real-world graphs into LLMs, owing to their context window constraints. This suggests that text-only encoding may not be the optimal approach for complex, large graph structures.

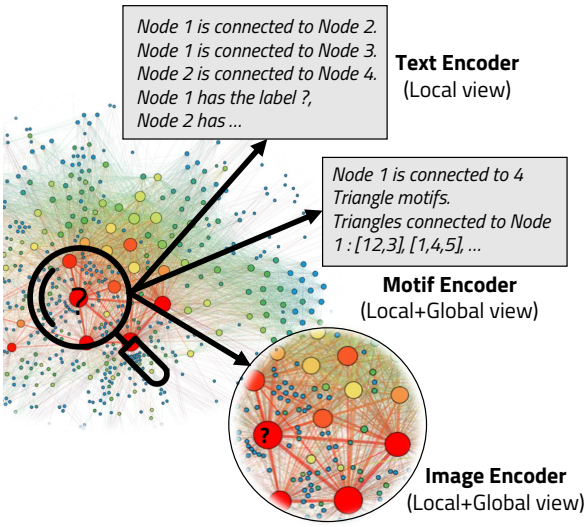


Figure 1: Input modality encoding for graphs impacts node classification, with text modality offering detailed information from a *local* point of view but violating the input context limitations for LLMs due to verbosity. Motif modality provides *local and global context*, while image modality gives a comprehensive *global* view, efficiently processed by GPT-4V, which integrates capabilities from both vision and text.

This paper investigates the impact of different modalities for encoding global and local graph structures, focusing on node classification tasks,

and compares three modalities: *Text*, *Motif*, and *Image* (See Figure 1). Text modality offers detailed local insights but becomes verbose for large graphs (Bubeck et al., 2023), often exceeding the input limits of models like GPT-4. The Motif modality is suggested to address this, capturing essential patterns in a node's vicinity for a balanced local-global perspective. Additionally, Image modality is proposed, utilizing fewer tokens to convey a more global view of the node's neighborhood, a method enhanced by the vision capabilities of the newly released GPT-4V (OpenAI, 2023a). Finding the optimal prompt input format is a notably complex challenge, with text modality encoding requiring extensive exploration compared to the simpler, more human-readable image modality. In our evaluations, we balance informativeness and prompt conciseness across all modalities using a combination of metrics.

Our **main contributions** are as follows:

- We conduct breadth-first analysis of various modalities, such as text, image, and motif, in graph-structure prompting, utilizing large language and vision-language models for node classification tasks.
- We also perform a depth-first analysis of how different factors influence the performance of each encoding modality.
- We introduce GRAPHTMI, a novel graph benchmark featuring a hierarchy of graphs, associated prompts, and encoding modalities designed to further the community's understanding of graph structure effects using LLMs.

Some key findings: 1) When balancing the constraint of token limits while preserving crucial information, the image modality is more effective than the text modality for graph-related tasks. 2) The choice of encoding modality for graph task classification depends on the task's difficulty, assessed by homophily and motif counts, with image modality being optimal for medium-difficulty tasks and motif modality for harder ones. 3) Factors like edge encoding function, graph structure, and graph sampling techniques impact the performance of node classification using text modality. 4) Motif attachment information has a more significant impact on node classification than motif count information. 5) Image representation correlated with human readability positively impacts node classification performance.

Our research shows that while LLMs are progressing in graph data processing, they still fall short compared to Graph Neural Networks in managing real-world graphs, emphasizing both the current limitations and the future potential of LLMs with image understanding in graph interpretation and reasoning tasks.

Properties	CORA	Citeseer	Pubmed
Classes	7	6	3
Nodes	2,708	3,327	19,717
Edges	5,278	4,552	44,324
Density	0.0014	0.0008	0.0002
Avg deg	3.89	2.74	4.49
Clust coeff	0.24	0.14	0.06
Diameter	∞	∞	18
Components	78	438	1
2-hop nodes	36	15	60

Table 1: Comparison of network properties of popular citation network datasets CORA, Citeseer and Pubmed.

2 Setups

2.1 Seed Datasets

We experiment with three citation network datasets, which are popular node classification benchmarks, CORA (McCallum et al., 2000) with seven categories : [0-Rule Learning, 1-Neural Networks, 2-Case-Based, 3-Genetic Algorithms, 4-Theory, 5-Reinforcement Learning, and 6-Probabilistic Methods], CITESEER (Giles et al., 1998) with six categories of areas in Computer Science: [0-Agents, 1-ML, 2-IR, 3-DB, 4-HCI, 5-AI] and PUBMED (Sen et al., 2008) that consists of scientific journals collected from the PubMed database with the following three categories: [0-Diabetes Mellitus, Experimental, 1-Diabetes Mellitus Type 1, 2-Diabetes Mellitus Type 2]. This paper focuses solely on the structural information of graphs for node classification. Hence, our experiments exclusively utilize node and label IDs.

2.2 Evaluation Metrics

This paper assesses the performance of node classification using four metrics chosen to balance the tradeoff between the encoding's informativeness and verbosity. The metrics used are Accuracy rate (which should increase ↑), Denial rate (which should decrease ↓), Mismatch rate (which should decrease ↓), indicating the prompt's informativeness, and Token limit fraction (which should de-

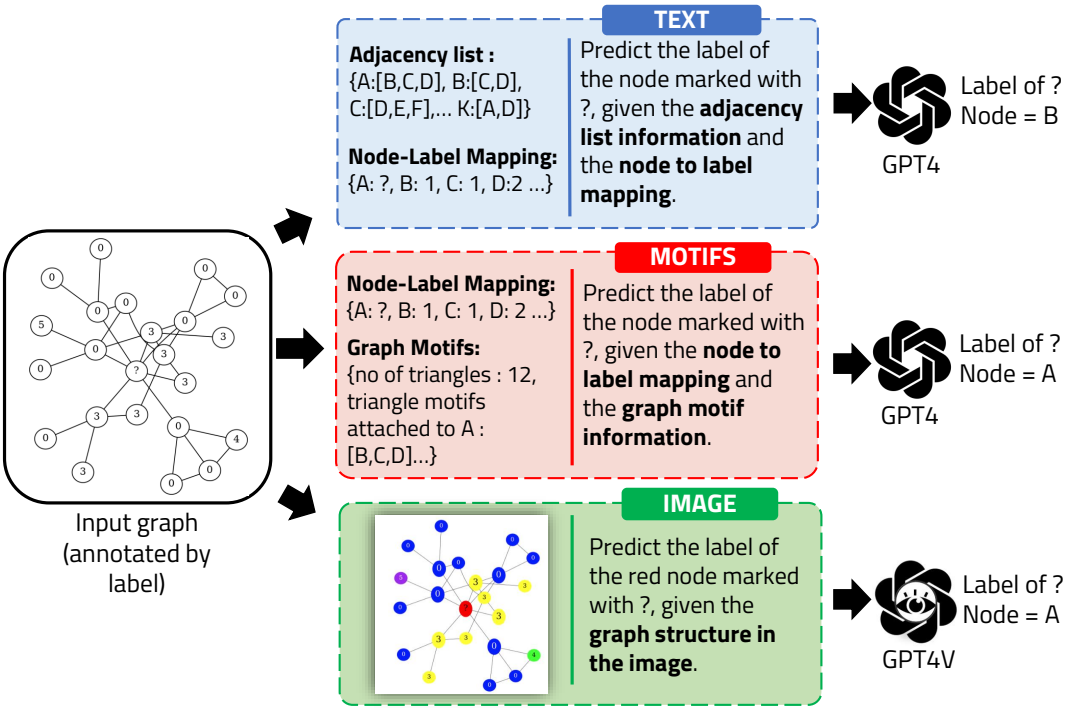


Figure 2: Node Classification on a Graph using different input modality encodings like Text, Motif, and Image.

crease ↓), reflecting the prompt's verbosity.

Accuracy Rate A: This metric indicates the LLM's performance on the task of node classification.

$$A = \frac{\text{No. of correct predictions}}{\text{Total no. of samples}} \quad (1)$$

Mismatch Rate M: This metric indicates the degree of misclassification by LLM (when the ground truth value is not the same as the predicted value).

$$M = \frac{\text{No. of incorrect predictions}}{\text{Total no. of samples}} \quad (2)$$

Denial Rate D: When we craft our prompt, we instruct the LLM to return -1 if it cannot predict the label of the ? node (node to be classified). The denial rate metric describes the rate of failure of the LLM (when the predicted value is -1).

$$D = \frac{\text{No. of predictions = -1}}{\text{Total no. of samples}} \quad (3)$$

$$1 - A = M + D \quad (4)$$

Token Limit Fraction T: This metric evaluates how effectively a Large Language Model's encoding modality uses its input context window, specifically focusing on the constraints imposed by the fixed-size attention window in transformer-based models like GPT-4 and GPT-4V. These constraints, dictated by the model's neural network architecture, limit the number of tokens that can be processed simultaneously, impacting both computational cost and performance.

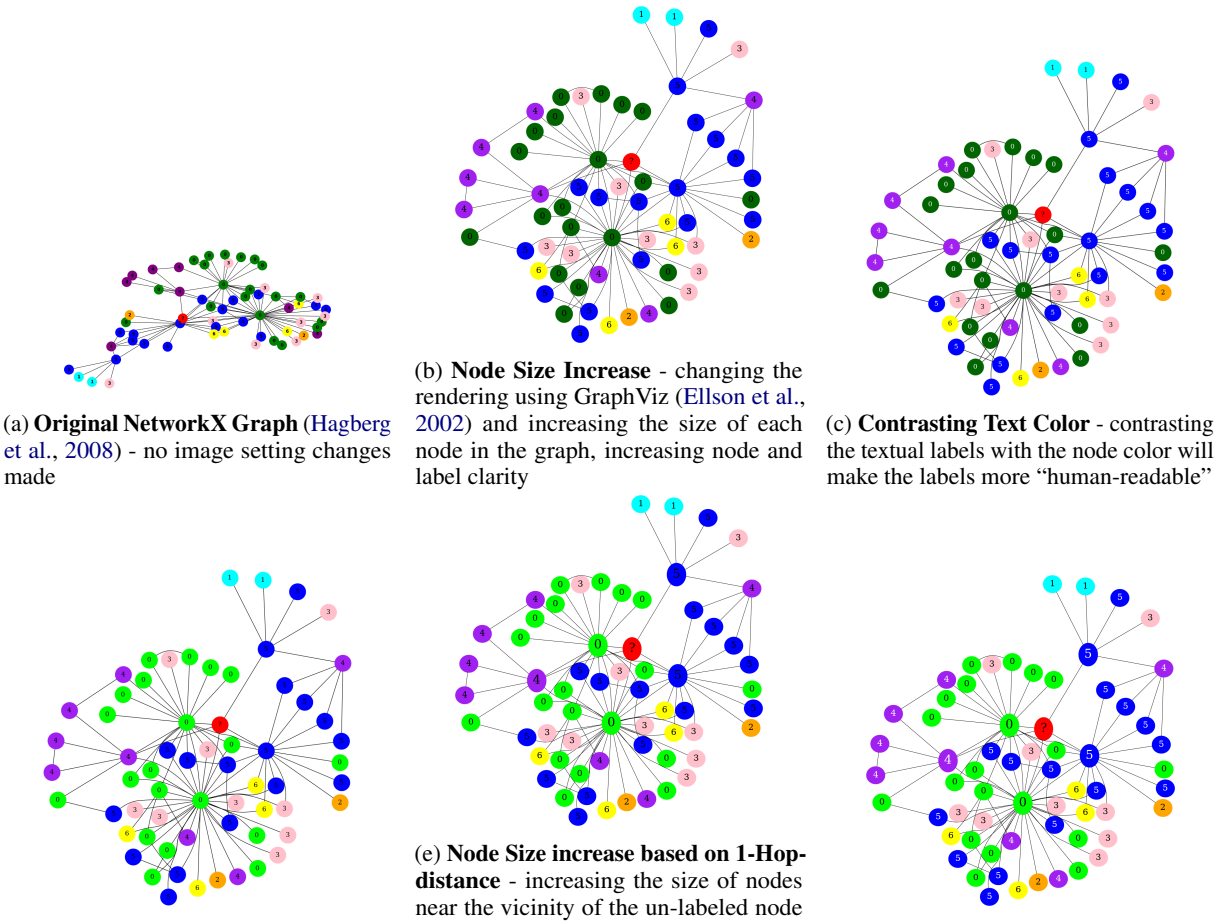
$$T = \frac{\text{Number of usage tokens}}{\text{Token limit constraint for the model}} \quad (5)$$

2.3 Graph Encoder Baselines

We compare our LLM models, which use different encoding modalities, to traditional graph learning models GCN(Kipf and Welling, 2016), GRAPHSAGE(Hamilton et al., 2017) and GAT (Veličković et al., 2017) as a baseline for node classification tasks. We provide the training details for the GNN models in Appendix B.

3 Proposed Encoders with Different Modalities

Graph encoding is crucial for converting graph-structured data into a sequence format that language models can process. As shown in Figure 2, the experimental setup involves using a modality encoder to input the graph structure and a graph query, such as predicting a node's label in node classification tasks. The graph structure is encoded according to the chosen modality (text and motif using GPT-4 and image using GPT-4V) and then



(a) **Original NetworkX Graph** (Hagberg et al., 2008) - no image setting changes made

(b) **Node Size Increase** - changing the rendering using GraphViz (Ellson et al., 2002) and increasing the size of each node in the graph, increasing node and label clarity

(c) **Contrasting Text Color** - contrasting the textual labels with the node color will make the labels more "human-readable"

(d) **Distinctive Node Colors** - choosing distinct colors allows the VLM to distinguish differently labeled nodes better

(e) **Node Size increase based on 1-Hop-distance** - increasing the size of nodes near the vicinity of the un-labeled node might allow the VLM to prioritize the data of nodes of greater size as opposed to nodes further away

(f) **Aggregate all changes** - all changes from b) to e) are applied to maximize image clarity

Figure 3: Image representation changes were applied sequentially on a graph, and we observed a distinct *increase from (a) to (f) in human readability and understanding of the graph* structure.

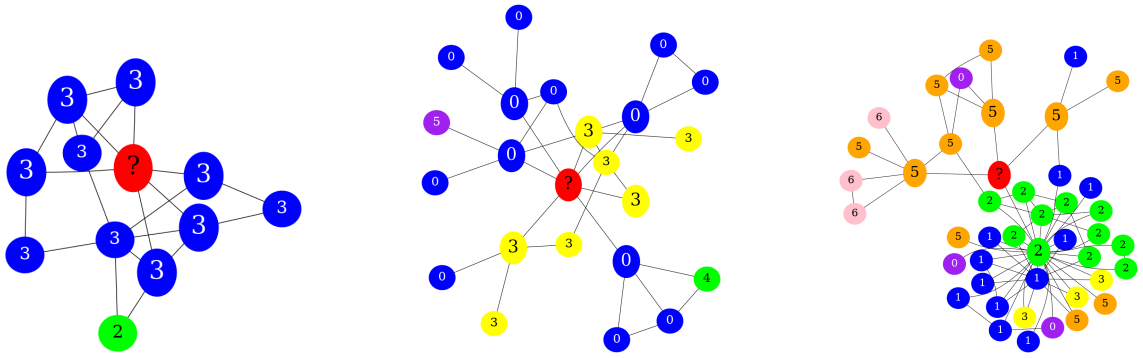
passed as a prompt to the LLMs to generate the required label.

3.1 Text Encoder

In encoding graphs as text, nodes are mapped to labels using a dictionary format, and different edge-encoding representations (Guo et al., 2023; Fatemi et al., 2023) are experimented with (Table 6), providing *local context* through edge connections and node labels to GPT-4 (OpenAI, 2023a). However, larger graphs can lead to verbose text encodings, which may exceed LLM input limits. We evaluate the impact of graph structure on classification (Yasir et al., 2023; Palowitch et al., 2022) by analyzing real-world citation datasets like PUBMED, CITESEER, and CORA, each with distinct network properties (definitions for these are provided in Table 8 and distinguished through Table 1). PUBMED is the largest and most connected but has the lowest clustering coefficient, indicating less local clustering. In contrast, CITESEER is highly fragmented with many disconnected components, while CORA, the smallest network, exhibits the highest density and clustering coefficient, suggesting strong local connectivity. Additionally, the research examines graph sampling techniques like ego graph (Stolz and Schlereth, 2021) and forest fire sampling (Leskovec and Faloutsos, 2006), crucial due to LLMs' limited context window and complex real-world graphs (Wei and Hu, 2022). These methods vary in their effectiveness, with Forest Fire sampling providing a broad network view, suitable for large networks like PUBMED, and Ego graph sampling excelling in revealing local community structures in more clustered and locally dense networks like CORA and CITESEER.

3.2 Motif Encoder

Network motifs, recurring patterns in social and biological networks (Milo et al., 2002; Carring-



(a) **EASY** - (#distinct labels < 3) and (#motifs ≤ 10)

(b) **MEDIUM** - ($3 \leq$ # distinct labels < 5) and ($10 <$ # motifs ≤ 20)

(c) HARD - (#distinct labels ≥ 5) and (# motifs ≥ 20)

Figure 4: Classifying graph task difficulty based on the criteria of Homophily and Number of Motifs yields a dataset of EASY, MEDIUM, and HARD graph problems and their associated modality encodings and classifications. This benchmark is called the GRAPHTMI dataset.

ton et al., 2005; Holland and Leinhardt, 1974), are pivotal in understanding local structures and behaviors. In LLMs, motif modality encoding leverages these motifs to provide *local and global context*, aiding in classifying unlabeled nodes (Yang et al., 2018). This process entails mapping nodes to labels using a dictionary format and identifying key motifs around the unlabeled node, which are inputted into GPT-4 as graph-motif information (detailed in Table 9). Differentiating between the count and specific members of motifs like stars, triangles, and cliques in a graph, our approach posits that a node's connection to influential motifs, such as being central in a star for network influence or part of a triangle or clique for close community ties, can significantly affect its classification by revealing key aspects of the network structure.

3.3 Image Encoder

Adopting the idea that "a picture is worth a thousand words", the image modality in graph analysis uses visual representations to outperform text in depicting structures, networks, labels, and spatial relationships using fewer tokens. Vision-language models like GPT-4V (OpenAI, 2023b) interpret these graph images, offering a *global context* of the graph's structure. GPT-4V, a multimodal model, merges visual interpretation with language processing, underscoring the importance of image representation in enhancing node classification. Our experiments involved using graph rendering methods to generate images with color-coded nodes, with a focus on improving human readability through various image modifications (Figure 3). These changes were evaluated for their impact on node classification, highlighting the critical role of visual representation in this modality.

4 GraphTMI Benchmark Creation

Our study reveals that the ease of node classification in graphs varies across different modalities, depending on the graph's "difficulty," determined by motif count and homophily. Homophily (McPherson et al., 2001), based on network theory, suggests that nodes are more likely to connect with similar nodes; thus, graphs with higher homophily (more nodes sharing the same label) are simpler to classify than those with more heterophily (diverse labels). This is illustrated through CORA dataset examples in Figure 4. Graphs are categorized as "easy," "medium," or "hard" based on the diversity of labels. Additionally, graphs with more network motifs are considered more complex and challenging for classification (Tu et al., 2018). The "task difficulty" is defined across eight categories ($2^3 = 8$), with the final difficulty level determined by the higher of the two criteria, homophily or motif count. This led to the creation of GRAPHTMI, a new benchmark dataset that includes various graph structures along with their respective modalities (text, motif, and image), prompts, and LLM classifications, thereby providing deeper insights into how different graphs affect LLM prompting. Specific statistics are given in Appendix A.

5 Results

5.1 Results Across All Modalities

Comparing Node Classification Accuracies between Graph baselines and LLM models : Table

	Model	Cora	Citeseer	Pubmed
GNN Baselines	GCN	0.7584 $\pm$0.121	0.6102 $\pm$0.087	0.7546 $\pm$0.076
	GAT	<u>0.7989</u> $\pm$0.092	0.6583 $\pm$0.074	0.7490 $\pm$0.060
	GraphSage	0.7719 $\pm$0.124	0.6017 $\pm$0.103	0.7193 $\pm$0.076
LLMs + Encoding Modality	Text	**0.81** $\pm$0.04 [0.07 $\pm$0.03]	**0.75** $\pm$0.05 [0.07 $\pm$0.01]	**0.83** $\pm$0.01 [0.08 $\pm$0.01]*
	Motif	0.73 $\pm$0.06 [0.06 $\pm$0.01]	0.59 $\pm$0.01 [0.32 $\pm$0.02]	0.77 $\pm$0.006 [0.13 $\pm$0.04]
	Image	0.77 $\pm$0.05 [0.04 $\pm$0.02]*	<u>0.71</u> $\pm$0.09 [0.06 $\pm$0.0]*	<u>0.79</u> $\pm$0.03 [0.19 $\pm$0.01]

Table 2: We report test accuracy rates of node classification across different datasets and denial rates D in [brackets] for LLM models. * indicates the lowest denial rate for each modality. The highest accuracy rate for the dataset is in bold, while the second highest is underlined. *The text modality in LLMS is comparable to GNN baselines with image modality not far behind.*

2 compares node classification accuracies of traditional GNN methods and LLM baselines across datasets, assessing if LLMs match conventional techniques. Limited by GPT-V's rate limit, the study used 50 ego graphs, with more extensive results in Appendix A. The *text modality of LLMs performs comparably to graph baselines in all datasets, with the image modality close behind, indicating LLMs' potential in graph analysis*. In larger datasets like PUBMED, the image modality showed a higher denial rate, possibly due to overcrowding in larger subgraphs, leading to more frequent classification denials by the LLM.

Comparing Node Classification Performance across Encoding Modalities: Figure 5 compares node classification across encoding modalities, focusing on accuracy, mismatch, denial rates, and token limit fraction. The text modality shows high accuracy but struggles with a high denial rate and token limit fraction, likely due to verbose inputs that confuse the LLM. In contrast, the image modality offers similar accuracy but with lower denial rates and token limit fractions, indicating the *image modality's effectiveness in providing a concise, global context that the LLM processes more efficiently*. This can be further emphasized through our qualitative analysis in Table 3.

Qualitative analysis of denial of classification in the Image Modality Figure 7 shows instances from multiple datasets where GPT-4V, using image modality, did not assign labels (returned -1) to graph nodes, explaining the reasons for denial. Key observations include: a) the LLM lacked *explicit context on label assignments* to nodes, as the encoding only implicitly indicated labels through node colors, with red reserved for unlabeled nodes. b) For one image, the absence of a clear *link between node colors and labels*, exacerbated by high heterophily, caused confusion. c) Another case highlighted the need for *few-shot learning*, suggesting that showing the LLM similar graph examples could help it learn to identify unlabeled (red) nodes more accurately.

Insights from GraphTMI In our evaluation of node classification accuracy using the GRAPHTMI benchmark across various modalities, we found in Figure 6 that "easy" tasks (characterized by high homophily and simpler structures) showed comparable accuracy across text, image, and motif modalities. However, for "medium" or "hard" tasks, marked by heterophyllous nature or complex structures, the image modality outperformed others, followed by the motif modality, underscoring the importance of global information in LLM processing. Notably, "hard" graphs achieved the highest accuracy with the motif modality, indicating the value of balancing local and global information. This suggests a *growing effectiveness of image and motif modalities in enhancing graph reasoning tasks like node classification*.

5.2 Modality Specific Results

Text modality results: Figure 16 shows that using the *Adjacency List* as the mode of edge representation with node label mapping is the *most informative encoding function*, which balances the trade-off between high accuracy and low token limit fraction. Figure 9 shows how metrics vary across datasets with different graph structures and sampling strategies. For CORA, a small, dense, and clustered graph, both sampling methods yield high accuracy, with forest fire (ff) sampling resulting in a lower denial rate. CITESEER, with its local clustering nature, struggles with ff sampling, showing the

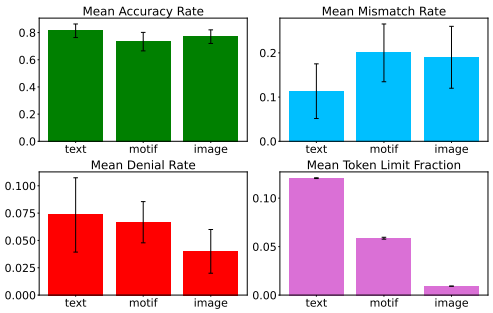


Figure 5: We observe that while the *text and image modalities have similar accuracy rates, the motif modality exhibits the highest mismatch rate, and the image modality stands out with the lowest denial rate and token limit fraction*, as depicted along the mean metrics (y-axis) against each modality type (x-axis)

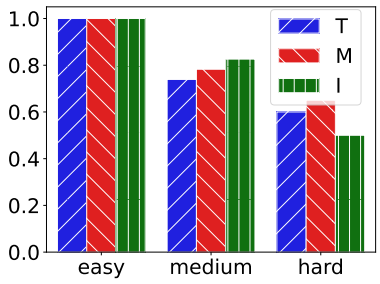


Figure 6: Modality encoder trends (T= Text, M= Motif, I= Image) with graph task difficulty based on homophily and no. of motifs, highlight the *significance of integrating local and global information in LLM processing.*

highest denial rate and the lowest mean accuracy, indicating difficulty in accurate predictions. In contrast, large and highly connected PUBMED generates larger samples through ego graph sampling, leading to higher token limit fractions. CITESEER's fragmented, disconnected nature results in smaller ego graph samples and lower token limit fractions. Thus, we can see *graph structure and sampling strategy significantly impact performance metrics*.

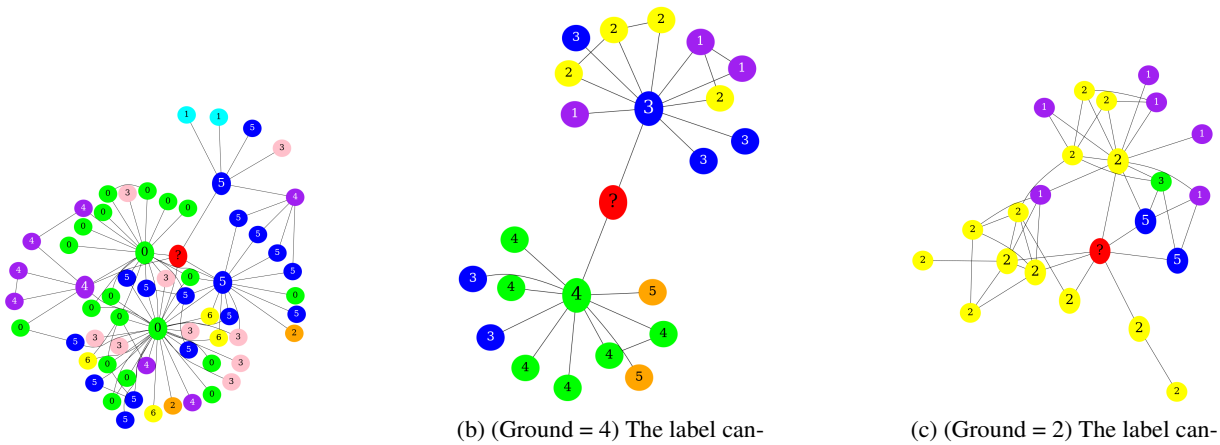
Motif modality results: Figure 17 shows GPT-4's *improved performance by adding the "triangle and star attached to ? node" motif* in the motif modality encoder (detailed in Appendix Table 9). This enhancement in mean accuracy and other metrics is attributed to the effective combination of local and global context provided to the LLM through node-label mapping and the associations within triads or star motifs.

Image modality results: Figure 3 shows different tweaks to image representation, and Figure 8 demonstrates that optimal node classification correlates with high accuracy and low denial and mismatch rates. Interestingly, *as human image readability increases, metric performance also improves, highlighting the easier use of images over text* for LLM prompts.

6 Related Work

LLMs with Graphs: Graph Neural Networks (GNNs) are renowned for their effectiveness in node classification and link prediction (Dwivedi et al., 2020), with applications in diverse fields like social networks, computer vision, and biology (Hou et al., 2022). GNNs struggle with processing non-numeric data like text and images, necessitating preprocessing such as feature engineering (Wang et al., 2021). In contrast, recent studies have explored using Large Language Models (LLMs) for graph reasoning, demonstrating their potential in complex tasks (Huang et al., 2022). This includes using LLMs for feature enhancement (Chen et al., 2023), node classification (Chen et al., 2023), and training neural networks in graph-based tasks (He et al., 2023), with benchmarks like NLGraph (Wang et al., 2023) assessing LLMs in traditional graph challenges. These studies typically employ LLMs as sub-components within graph learning frameworks. *Our research examines LLMs' ability to process graph modalities directly, aiming to understand LLMs' intrinsic graph-handling capabilities, thus presenting a novel direction in the field.*

Prompt Design for Graphs: Prompting strategies for querying large language models (LLMs) aim to optimize the prompt text for enhanced task performance. Few-shot in-context learning (Brown et al., 2020) provides examples with desired outputs for the model to learn and generalize. Chain-of-thought (CoT) prompting (Wei et al., 2022) offers step-by-step problem-solving examples, leading the model to develop reasoning paths, while its zero-shot variant (Kojima et al., 2022) initiates reasoning with a starter phrase. Bag prompting (Wang et al., 2023) focuses on graph tasks, recommending graph construction before the task. Format explanations and role prompting (Guo et al., 2023) are proposed for better task clarity and strategic input organization to leverage LLMs' learning capabilities. Self-prompting involves the LLM refining



(a) (Ground = 0) Without additional context or rules for how labels are assigned, it is not possible to accurately predict the label of the red node.

(b) (Ground = 4) The label cannot be determined with certainty due to the lack of a discernible pattern or rule that associates a node's color or its connections with its label.

(c) (Ground = 2) The label cannot be determined with certainty due to the lack of a clear pattern in the graph and no previous examples of red nodes to infer from.

Figure 7: Examples of graphs where VLM (GPT-4V) returned -1 or denied to predict a label and the reason for denial. The ground truth for this graph is given in brackets. This *highlights the need to clarify labeling strategies and few shot learning.*

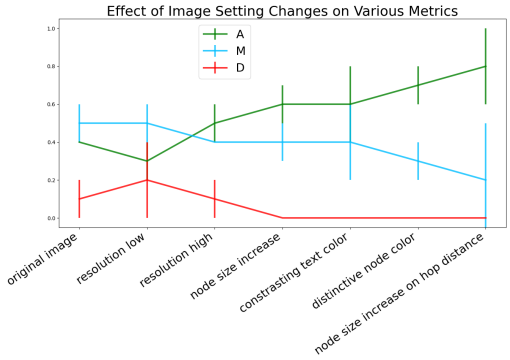


Figure 8: Our comparison of image representations (x-axis) with mean metrics (y-axis) shows that *human readability of images correlates with classification performance*, considering Accuracy Rate (A ↑), Mismatch Rate (M ↓), and Denial Rate (D ↓), with desired trends indicated in brackets.

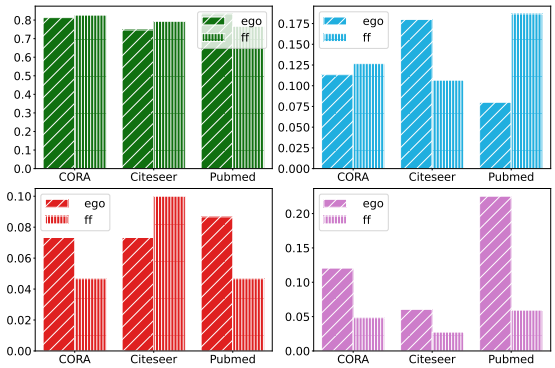


Figure 9: Each dataset and sampling type (x-axis) is mapped against mean metrics (y-axis), with bar textures distinguishing between ego graph (ego) and forest fire (ff) sampling; the metrics include accuracy rate (↑), mismatch rate (↓), denial rate (↓), and token limit fraction (↓), indicating the desired trends for each. *Graph structures of different datasets and sampling strategies influence node classification performance.*

prompts via context summarization, tackling issues with complex or insufficient graph data. *Our study employs zero-shot prompting, providing only a task description to the LLM, to concentrate on the impact of modalities without the influence of varied prompt designs.*

7 Conclusion and Future Work

This study explores the application of LLMs in graph-structured data, evaluating their strengths and weaknesses in node classification using various input modalities like motif and image for effective data representation. Introducing the GRAPHTMI benchmark highlights the image modality's efficiency in token limit management. Although LLMs have progressed in graph data processing, they still don't match the performance of GNNs in practical settings. The research advocates for future work combining different modalities to improve node classification, combining LLM-based methods with GNNs, applying these techniques to complex, text-dense graphs, and delving into link prediction and community detection to broaden applications and insights across multiple domains.

Limitations

Although innovative in applying LLMs to graph-structured data, this research faces key limitations. The computational demands of detecting network motifs, essential for understanding complex network dynamics, pose a significant challenge. This process requires extensive computational power and advanced algorithms, limiting scalability and efficiency. We subvert these challenges by restricting our subgraph sample size to 3 hops of an ego graph. Additionally, the study is constrained by the current rate limitations of GPT-V, which affects its ability to process and analyze data at scale efficiently. This limitation restricts the depth and scope of the analysis, although future advancements in GPT-V may mitigate this issue. A further limitation lies in the study's simplistic approach to estimating homophily, relying merely on label count and neglecting the importance of hop distance. This overlooks critical network structure and node similarity aspects, leading to a potentially oversimplified analysis. Incorporating hop distance could provide a more accurate representation of network homophily. These limitations underscore the need for further advancements in computational techniques, model capabilities, and more nuanced theoretical methods in network analysis.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Peter J Carrington, John Scott, and Stanley Wasserman. 2005. *Models and methods in social network analysis*, volume 28. Cambridge university press.

Zhikai Chen, Haitao Mao, Hang Li, Wei Jin, Hongzhi Wen, Xiaochi Wei, Shuaiqiang Wang, Dawei Yin, Wenqi Fan, Hui Liu, et al. 2023. Exploring the potential of large language models (llms) in learning on graphs. *arXiv preprint arXiv:2307.03393*.

Vijay Prakash Dwivedi, Chaitanya K Joshi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. 2020. Benchmarking graph neural networks. *arXiv preprint arXiv:2003.00982*.

John Ellson, Emden Gansner, Lefteris Koutsofios, Stephen C North, and Gordon Woodhull. 2002. Graphviz—open source graph drawing tools. In *Graph Drawing: 9th International Symposium, GD 2001 Vienna, Austria, September 23–26, 2001 Revised Papers 9*, pages 483–484. Springer.

Bahare Fatemi, Jonathan Halcrow, and Bryan Perozzi. 2023. Talk like a graph: Encoding graphs for large language models. *arXiv preprint arXiv:2310.04560*.

C Lee Giles, Kurt D Bollacker, and Steve Lawrence. 1998. Citeseer: An automatic citation indexing system. In *Proceedings of the third ACM conference on Digital libraries*, pages 89–98.

Jiayan Guo, Lun Du, and Hengyu Liu. 2023. Gpt4graph: Can large language models understand graph structured data? an empirical evaluation and benchmarking. *arXiv preprint arXiv:2305.15066*.

Aric Hagberg, Pieter Swart, and Daniel S Chult. 2008. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

Xiaoxin He, Xavier Bresson, Thomas Laurent, and Bryan Hooi. 2023. Explanations as features: Llm-based features for text-attributed graphs. *arXiv preprint arXiv:2305.19523*.

Paul W Holland and Samuel Leinhardt. 1974. The statistical analysis of local structure in social networks.

Yifan Hou, Jian Zhang, James Cheng, Kaili Ma, Richard TB Ma, Hongzhi Chen, and Ming-Chang Yang. 2022. Measuring and improving the use of graph information in graph neural networks. *arXiv preprint arXiv:2206.13170*.

Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. 2022. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International Conference on Machine Learning*, pages 9118–9147. PMLR.

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

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Jure Leskovec and Christos Faloutsos. 2006. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 631–636.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Andrew Kachites McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. 2000. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3:127–163.

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

Prem Melville and Vikas Sindhwani. 2010. Recommender systems. *Encyclopedia of machine learning*, 1:829–838.

Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. 2002. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827.

Alan Mislove, Massimiliano Marcon, Krishna P Gummadi, Peter Druschel, and Bobby Bhattacharjee. 2007. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 29–42.

OpenAI. 2023a. Gpt-4 system card. https://openai.com/research/gpt-4v-system-card.

OpenAI. 2023b. Gpt-4v(ision) system card. https://cdn.openai.com/papers/GPTV_System_Card.pdf.

John Palowitch, Anton Tsitsulin, Brandon Mayer, and Bryan Perozzi. 2022. Graphworld: Fake graphs bring real insights for gnns. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3691–3701.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2023. Unifying large language models and knowledge graphs: A roadmap. *arXiv preprint arXiv:2306.08302*.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. 2008. Collective classification in network data. *AI magazine*, 29(3):93–93.

Simon Stolz and Christian Schlereth. 2021. Predicting tie strength with ego network structures. *Journal of Interactive Marketing*, 54(1):40–52.

Kun Tu, Jian Li, Don Towsley, Dave Braines, and Liam D Turner. 2018. Network classification in temporal networks using motifs. *arXiv preprint arXiv:1807.03733*.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Saraswathi Vishveshwara, KV Brinda, and Natarajan Kannan. 2002. Protein structure: insights from graph theory. *Journal of Theoretical and Computational Chemistry*, 1(01):187–211.

Heng Wang, Shangbin Feng, Tianxing He, Zhaoxuan Tan, Xiaochuang Han, and Yulia Tsvetkov. 2023. Can language models solve graph problems in natural language? *arXiv preprint arXiv:2305.10037*.

Yangkun Wang, Jiarui Jin, Weinan Zhang, Yong Yu, Zheng Zhang, and David Wipf. 2021. Bag of tricks for node classification with graph neural networks. *arXiv preprint arXiv:2103.13355*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Qiang Wei and Guangmin Hu. 2022. Evaluating graph neural networks under graph sampling scenarios. *PeerJ Computer Science*, 8:e901.

Carl Yang, Mengxiong Liu, Vincent W Zheng, and Jiawei Han. 2018. Node, motif and subgraph: Leveraging network functional blocks through structural convolution. in 2018 ieee. In *ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 47–52.

Mustafa Yasir, John Palowitch, Anton Tsitsulin, Long Tran-Thanh, and Bryan Perozzi. 2023. Examining the effects of degree distribution and homophily in graph learning models. *arXiv preprint arXiv:2307.08881*.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2023. A survey on multimodal large language models. *arXiv preprint arXiv:2306.13549*.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

A LLM Experiments

A.1 Qualitative Analysis on Combining Modalities

We perform a qualitative analysis of the response returned by LLMs by utilizing the text, image,

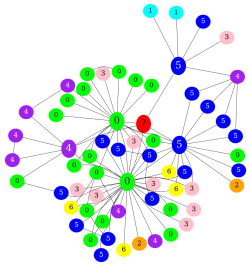
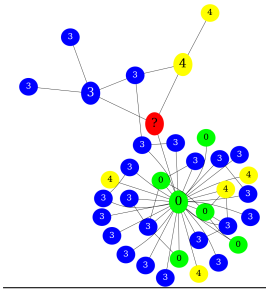
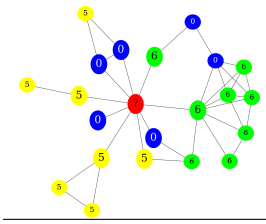
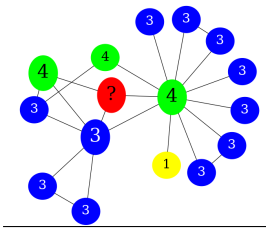
Image	Ground Truth	Text_Response	Text + Image_Reponse	Image_Response
	*Label of Node = 5*	The label of a node can often be predicted based on the labels of its neighboring nodes. In this case, the node in question is '1677', and its neighbors are '88', '415', '1309', and '191', '1908'. There is no clear majority label among these neighbors. Therefore, it is not possible to predict the label of node '1677' based on this information. *Label of Node = -1*	From the adjacency list, we can determine that the neighbors of node "1677". Given that node "1677" is connected to two nodes with label 0, two nodes with label 5, and one node with label 4, we can attempt to predict its label based on the labels of its neighbors. Since nodes "1309" and "191" both have the label 5 and are each directly connected to "1677," it's reasonable to refer to the homophily in networks concept where nodes are likely to be of the same type (label) as their neighbors. Based on this information, it seems plausible that the red node "1677" would also have the label that appears most frequently among its neighboring nodes, which is label 5. *Label of Node = 5*	*Label of Node = 0*
	*Label of Node = 3*	The node 1329 is connected to nodes 733, 994, and 78. The labels of these nodes are 0, 3, and 4 respectively. Since there is no majority label among these, the label of node 1329 cannot be determined. *Label of Node = -1*	*Label of Node = 3*	*Label of Node = 3*
	*Label of Node = 6*	The node 702 is connected to nodes 777, 779, 2069, 2070, 2101, 822, 993, 1368, and 1637. The labels of these nodes are 0, 6, 0, 0, 5, 5, 6, 5, and 0 respectively. The most common label among these is 0. Therefore, we can predict that the label of node 702 is also 0.*Label of Node = 0*	*Label of Node = 6*	*Label of Node = 6*
	*Label of Node = 3*	The node 141 is connected to nodes 1002, 2034, and 740. The labels of these nodes are 3, 4, and 4 respectively. Since the majority of the connected nodes have the label 4, we can predict that the label of node 141 is also 4. *Label of Node = 4*	*Label of Node = 3*	*Label of Node = 3*

Table 3: The table shows the GPT-4 and GPT4-V responses with the text modality, text+image modality, and image modality. *We observe that on combining modalities, the label originally classified as -1 is correctly classified (first two rows), or the original misclassification is rectified (last two rows).*

and text combined with image encoding modalities. The intuition here is that the local context provided by the text modality might not be enough for some predictions and could be supplemented through the global context provided by the image modality. *Table 3 illustrates that misclassifications and denials by the LLM using text modality could be rectified by using the image modality*. For the first two rows, -1 classifications or LLM denials are changed to the correct classification on incorporating the global context of the image modality. We can see in the response that the notion of "homophily" is clearer to the VLM in the image modality. For the last two rows, we see that the graph is originally misclassified, but then this is corrected by incorporating the image modality. We make similar observations on combining text and motif modalities, and this could be because another factor important to node classification is the presence of motifs, which is highlighted through the motif modality.

A.2 Comparing encoding modalities for different datasets and sampling

Our study evaluates various encoding modalities — text, motif, and image — with ego graphs from CORA detailed in the main manuscript. Figure 15 extends this analysis to the other datasets and sampling techniques. The findings corroborate our assertion that graph structure and the chosen sam-

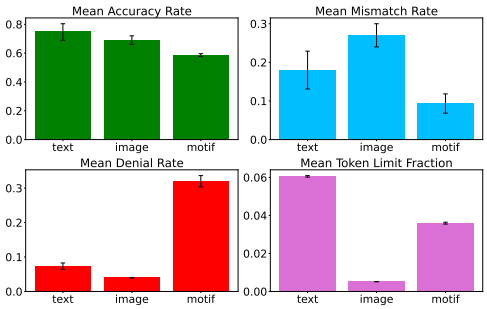


Figure 10: Citeseer with ego graph sampling

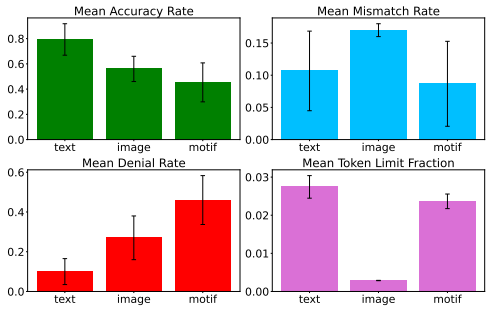


Figure 11: Citeseer with forest fire sampling

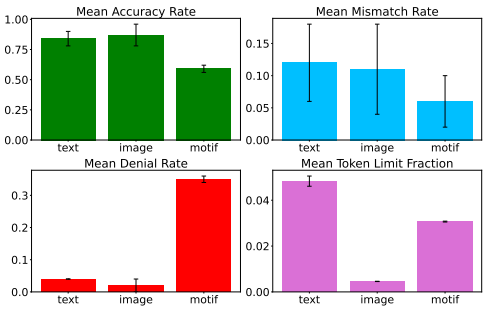


Figure 12: Cora with forest fire sampling

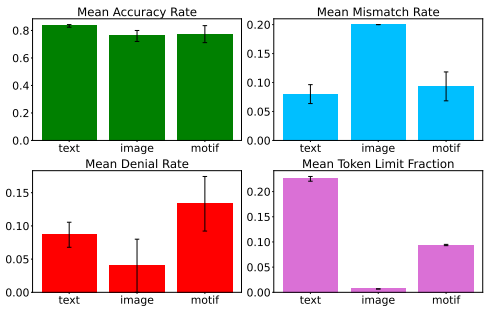


Figure 13: Pubmed with ego graph sampling

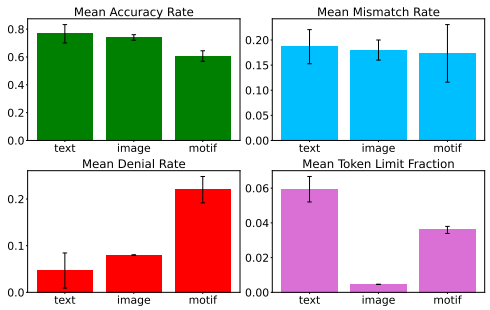


Figure 14: Pubmed with forest fire sampling

Figure 15: Modality comparison (text, motif, and image) with the graph structure and sampling type shows the *clear dependency of graph structure and sampling on node classification performance*.

pling method significantly influence node classification outcomes. Particularly, samples derived from the forest fire method, which emphasize the global configuration while being sparser and less connected than ego graphs, exhibit increased misclassification rates when using the image modality due to limited information for accurate inference and greater instances of non-committal predictions with the motif modality due to the absence of a discernible overarching structure.

A.3 Graph TMI Benchmark

We decide on graph "difficulty" based on the dual criteria of 1) count of motifs and 2) homophily in the graph. We apply a naive heuristic to decide homophily, i.e., the count of the distinct labels in the graph. If the count of distinct labels < 3, the graph is considered *easy*. If the count is ≥ 3 and < 5, it is considered *medium*, and if the count is ≥ 5, it is considered *hard*. To decide the motif criteria, we count the total number of motifs(focusing on just triads, star motifs, and cliques) in a graph. For example, if this count of motifs ≤ 10 for the CORA dataset, the graph is considered *easy*. If the count is > 10 and ≤ 20, it is regarded as *medium*; if the count is > 20, it is considered *hard*. Some graphs can have both the homophily and motif criteria applicable to them; for instance, Figure 4 (b) can be classified as *medium* based on homophily, *hard* based on the count of motifs. This leads us to combine the homophily and the count of motif criteria to define the "task difficulty". Thus, we can have $2^3 = 8$ categories of difficulty, and the final difficulty label is decided by choosing the

higher annotation between homophily and count of motif classification). So, a graph with criteria {easy, hard} will be assigned the final task difficulty, *hard*. Thus, we can classify graphs based on our "task difficulty" heuristic, and we introduce GRAPHTMI (Graph Text-Motif-Image), a novel benchmark dataset of input graph structures paired with their associated modality encodings.

	homophily	motif	count
0	easy	easy	7
1	easy	hard	4
2	easy	medium	6
3	hard	easy	1
4	hard	hard	8
5	hard	medium	2
6	medium	easy	5
7	medium	hard	5
8	medium	medium	12

Table 4: Statistics about the number of graphs classified as easy, medium, or hard through the homophily and the number of motifs criteria. All possible combinations are covered in our benchmark ($3^2 = 9$).

	difficulty	count
0	easy	7
1	hard	20
2	medium	23

Table 5: Statistics about the number of problems finally classified as easy, medium, or hard based on task difficulty, a function of homophily, and number of motifs.

A.4 Modality Specific Experiment Details

Token limits for each modality Due to their architecture, transformer-based models like GPT-3 and GPT-4 have a fixed-size attention window. This determines how many tokens the model can "remember" or pay attention to at once. This limit also manages the computational cost of running the model and the model's performance. The token limit constraint for GPT-4 is 8192 tokens, while for GPT-4V(vision), the limit is claimed to be 128000 tokens, but currently, only the preview version has been released, and the actual limit is 10000 tokens.

Rate limits for each LLM The rate limit for GPT-4 is 10K RPM (requests per minute), and for GPT-4V, the rate limit is 100 RPD (requests per day).

Modality Experiment Parameters For all modality types, we sample 50 graphs for all datasets, the number of hops considered = 3, no of runs = 2, and perform ego graph and forest fire sampling. We report the mean and standard deviation directly or through error bars in the visualization for all metrics. In the paper, we report the results from ego graph sampling because node classification typically needs a localized view around specific nodes, best provided by ego graph sampling.

A.4.1 Text Modality

Encoding graphs as text can be separated into two key parts: First, the mapping of nodes to their corresponding labels in the graph, and second, the encoding of edges between the nodes. We encode the node-to-label mapping as a dictionary of type {node ID: node label}. Finding a concise yet informative representation of the graph structure and edge representation is essential. Example of a prompt using text modality :

Task: Node Label Prediction (Predict the label of the node marked with a ?) given the adjacency list information as a dictionary of type "node: neighborhood" and node-label mapping in the text enclosed in triple backticks. The response should be in the format "Label of Node = <predicted label>". If the predicted label cannot be determined, return "Label of Node = -1".
```**AdjList**: {1: [2,3], 2: [3,4], 3: [1,2]}
**Node-Label Mapping**: {1: A, 2: B, 3: ?} ```

**Impact of Edge encoding function**: Motivated by recent works (Fatemi et al., 2023; Guo et al., 2023) describing the importance of selecting the appropriate text encoding for a graph, we experiment with different edge representations (Appendix Table 6) on real-world datasets and evaluate the metrics for node classification and the results of this are illustrated in Figure 16. *"Adjacency list" is the best-performing edge representation for the text modality.*

**Impact of Graph Structure**: We selected diverse real-world citation datasets with unique network characteristics, as shown in Table 1. These network properties are defined in Table 8. The average number of nodes and edges in a 2-hop subgraph is also reported for CORA, Citeseer, and Pubmed datasets.

**Impact of Sampling Strategy**: Graph sampling techniques are essential for applying LLMs in graph reasoning, particularly due to the limited context window of LLMs and the intricacy of real-world graphs (Wei and Hu, 2022). Ego graph sampling centers on a specific node and its direct connections, forming a subgraph that mirrors these immediate relationships. In contrast, Forest Fire sampling randomly selects a node and expands from there, producing varying subgraphs in size
```

Edge Representation	Text Encoding	Description of Edge Representation
Edgelist	Node to Label Mapping : Node 69025: Label 34| Node 17585: Label 10|... *Edge list: [(69025, 96211), (69025, 17585), (17585, 104598), (17585, 18844), (17585, 96211), (96211, 34515)]*	An Edgelist is a graph data structure that represents a graph by listing the edge connections between two nodes. (A, B) indicates an connection between nodes A and B.
Edgetext	Node to Label Mapping : Node 85328: Label 16| Node 158122: Label ?|... *Edge connections (source node - target node): Node 85328 is connected to Node 158122. Node 158122 is connected to Node 167226.*	An Edgetext explicitly lists the connections between two nodes; for example, Node A is connected to Node B or Node A - Node B
Adjacency List	Node to Label Mapping : Node 2339: Label 3| Node 2340: Label ?|... *Adjacency list: 1558: [2339, 2340], 2339: [1558, 2340], 2340: [2339, 1558]*	An adjacency list represents a graph as an array of linked lists. The index of the array represents a vertex, and each element in its linked list represents the other vertices that form an edge with the vertex. For example, A: [B, C] shows that A is connected to B and C. This gives an idea of node-neighborhood
GML	GraphML: graph [node [id 2339 label 3] node [id 2340 label ?] node [id 1558 label 3] edge [source 2339 target 1558] edge [source 2339 target 2340]]	A GraphML format consists of an unordered sequence of node and edge elements enclosed within []. Each node element has a distinct id and label attribute contained within []. Each edge element has source and target attributes contained within [] that identify the endpoints of an edge by having the same value as the node id attributes of those endpoints. The node label information is embedded within the structure, meaning no node-label mapping is notneeded.
GraphML	GraphML: <graphml xmlns=http://graphml.graphdrawing.org/xmlns xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance xsi:schemaLocation=http://graphml.graphdrawing.org/xmlns http://graphml.graphdrawing.org/xmlns/1.0/graphml.xsd> <graph edgedefault=undirected> <node id=2339 label=3 /> <node id=2340 label=? /> <node id=1558 label=3 /> <edge source=2339 target=1558 /> <edge source=2339 target=2340 /> </graph> </graphml>	A GraphML file consists of an XML file containing a graph element, within which is an unordered sequence of node and edge elements. Each node element should have a distinct id attribute as well as its label, and each edge element has source and target attributes that identify the endpoints of an edge by having the same value as the id attributes of those endpoints. The node label information is embedded within the structure meaning no node-label mapping is needed.

Table 6: Summary of edge representation passed as a part of the text modality encoder with their associated examples and explanations. We find that the **Adjacency list** representation provides a granular yet not too verbose view of the graph being passed to the LLM.

	CORA	Citeseer	Pub.med
Avg edges 2-hop	62.70 ± 94.77	26.35 ± 61.70	129.36 ± 287.61
Avg nodes 2-hop	36.78 ± 48.12	15.11 ± 24.73	60.05 ± 85.12

Table 7: Subgraph Sampling Statistics about average number of nodes and edges in a 2-hop subgraph from each dataset.

and structure, influenced by factors like 'burning' probabilities. However, both methods have limitations and can potentially distort the overall structure of complex and extensive networks.

A.4.2 Motif Modality

Encoding graphs as motifs can be separated into two key parts: First, the encoding of nodes to their corresponding labels in the graph, and second, the motifs present around the ? (unlabeled) node. We encode the node-to-label mapping as a dictionary of type {node ID: node label}. We calculate motifs in the neighborhood of the ? nodes and pass this information to GPT-4 (OpenAI, 2023a) as the graph-motif information. Connections of an unlabeled node to significant nodes or groups (like stars or cliques) are more indicative of its label than just the count of graph motifs, with central nodes in star motifs or members of cliques heavily influenced by their neighbors' labels. We experiment with different network motifs as input to the modality encoder. Table 9 describes the different types of motifs considered, a description of the motif, and an example of the encoding generated as input to GPT-4. An example prompt generated after applying the motif

Table 8: Graph Properties and Their Descriptions

Name of Property	Description
Density	Measures how connected the graph is. It's the ratio of actual edges to possible edges.
Degree Distribution	The distribution of node degrees. The histogram might follow a specific pattern (e.g., power-law distribution, Gaussian distribution).
Average Degree	The average degree of nodes in the graph.
Connected Components	A subgraph in which a path connects any two nodes.
Clustering Coefficient	Measures the degree to which nodes tend to cluster together.
Graph Diameter	The longest shortest path between any two nodes. It provides insight into the graph's overall size.
2hop nodes	Average number of nodes present in the subgraph at 2 hop distance from any node

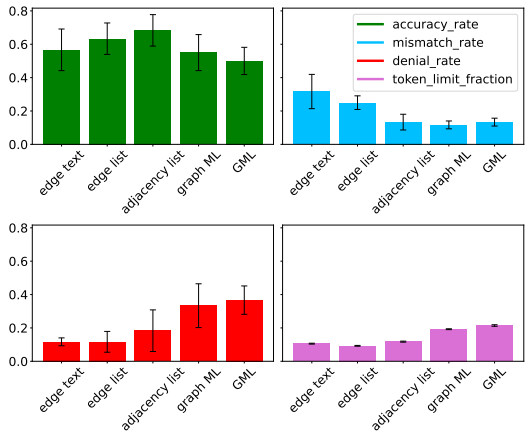


Figure 16: We compare the edge representation type (x-axis) with the value of the mean metrics (y-axis). The desired trend is given in brackets for each metric. The highest performing edge representation is the "adjacency list" representation with the highest accuracy (A ↑) and low mismatch rate (M ↓)), denial rate (D ↓), and token limit fraction (T ↓).

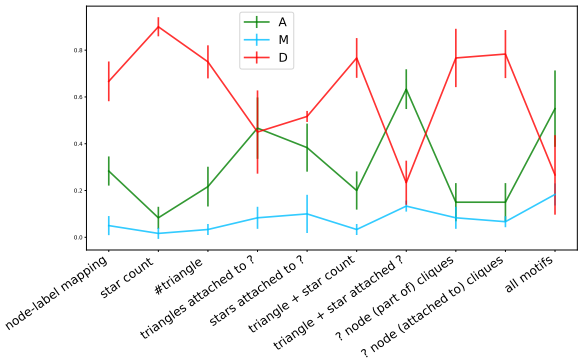


Figure 17: We compare the motif information (x-axis) to the mean metrics (y-axis). Desired trends are denoted in brackets. Metrics considered are Accuracy Rate (A ↑), Mismatch Rate (M ↓), and Denial Rate (D ↓). The highest performing motif information change "triangle and star attached to ?" has higher accuracy and lower mismatch and denial rate.

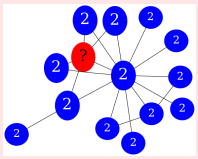
encoding modality looks like:

Task: Node Label Prediction (Predict the label of the node marked with a ?) given the node-label mapping and graph motif information in the text enclosed in triple backticks. The response should be in the format "Label of Node = <predicted label>". If the predicted label cannot be determined, return "Label of Node = -1".
```**Node-Label Mapping**: {1: A, 2: A, 3: ?}
**Graph-motif information**: No of triangles: 1| Triangles attached to ? Node : [1,2,3]| ```

### A.4.3 Image Modality

We use GPT-4V (OpenAI, 2023b) to process graph images to give LLMs a global perspective of graph structural information. An example prompt generated after applying the image encoding modality is shown below.

**Task**: Node Label Prediction (Predict the label of the red node marked with a ?, given the **graph structure information in the image**). The response should be in the format "Label of Node = <predicted label>." If the predicted label cannot be determined, return "Label of Node = -1."



| Details | GCN | GAT | GraphSAGE |
|---|---|---|---|
| Epochs | 100 | 100 | 100 |
| Learning Rate | 0.005 | 0.005 | 0.005 |
| Weight Decay | 5e-4 | 5e-4 | 5e-4 |

Table 11: List of GNN training hyperparameters

## B GNN Experiments

For training our GNN models, we used the best-found hyperparameters reported in Table 11. The
```

Type of Motif	Motif Encoding	Description of Motif
Node-Label Mapping	*Node to Label Mapping : Node 1889: Label 4* | ... *Node 1893: Label ?*|...	Only the node-label mapping is provided (this gives no connectivity information to LLM)
No. of Star Motifs	Node to Label Mapping : Node 1889: Label 4 | ... Node 1893: Label ?| ... *Graph motif information: Number of star motifs: 0*|	Star motifs signify centralized networks with influential central nodes, where a central node is connected to others that aren't interlinked. We pass the count of the star motifs present in the graph.
No. of Triangle Motifs	Node to Label Mapping : Node 1889: Label 4 | ... Node 1893: Label ?| ... *Graph motif information: Number of triangle motifs: 6*|	Triangle motifs (triads connecting three nodes) are foundational in social networks, indicating transitive relationships, community structures, and strong social ties. We pass the count of the triads present in the graph.
No. of Triangle Motifs Attached	Node to Label Mapping : Node 1889: Label 4 | ... Node 1893: Label ?|... *Graph motif information: Triangle motifs attached to ? node: [1893,2034,1531], [1893,1531,429]*|	We pass the triangle motifs attached to the ? label, which gives an idea of the influential triads connected to the ? node.
No. of Star Motifs Attached	Node to Label Mapping : Node 1889: Label 4 | ... Node 1893: Label ?|... *Graph motif information: Star motifs connected to ? node: []*|	We pass the star motifs attached to the ? label, which gives an idea of the influential nodes connected to the ? node.
No. of Star and Triangle Motifs	Node to Label Mapping : Node 1889: Label 4 | ... Node 1893: Label ?|... *Graph motif information: Number of star motifs: 0*| *Number of triangle motifs: 6*|	We pass the count of the triads and star motifs present in the graph, to give the LLM an idea of the graph structure.
Star and Triangle Motifs attached	Node to Label Mapping : Node 1889: Label 4 | ... Node 1893: Label ?|... *Graph motif information: Triangle motifs attached to ? node: [1893,2034,1531], [1893,1531,429]*| *Star motifs connected to ? node: []*|	We pass the star motifs and triads attached to the ? label, which gives an idea of the influential nodes and triads connected to the ? node.
No of cliques ? Node is part of	Node to Label Mapping : Node 1889: Label 4 | ... Node 1893: Label ?|... *Graph motif information: Number of cliques in graph: 0*| *? Node is a part of these cliques: []*|	We pass the number of cliques in the network, which gives an idea of its clustered nature. We also pass the cliques the ? label is a part of, which gives an idea of the immediate community of the unlabelled node.
No of cliques ? Node is attached to	Node to Label Mapping : Node 1889: Label 4 | ... Node 1893: Label ?|...*Graph motif information: ? Node is attached to these cliques: []*|	We pass the cliques the ? label is attached to, which gives an idea of the neighboring influential community of the unlabelled node.

Table 9: Summary of motif information passed as a part of the motif modality encoder with their associated examples and explanations. The **Aggregate of all changes** setup combines all of the above motif information to give the LLM a local and global view of the graph being passed.

	Model	Cora	Citeseer	Pubmed
GNN Baselines	GCN	0.7820 ±0.133	0.6540 ±0.083	0.7480 ±0.077
	GAT	**0.8200** ±0.084	0.6680 ±0.069	0.7510 ±0.050
	GraphSage	0.7570 ±0.137	0.6300 ±0.098	0.7430 ±0.078
LLMs + Encoding Modality	Text	0.81 ±0.04 [0.07 ±0.03]	**0.75** ±0.05 [0.07 ±0.01]	**0.83** ±0.01 [0.08 ±0.01]*
	Motif	0.73 ±0.06 [0.06 ±0.01]	0.59 ±0.01 [0.32 ±0.02]	0.77 ±0.006 [0.13 ±0.04]
	Image	0.77 ±0.05 [0.04 ±0.02]*	0.71 ±0.09 [0.06 ±0.0]*	0.79 ±0.03 [0.19 ±0.01]

Table 10: Test accuracy rates of node classification across different datasets using the entire 1000 test data and denial rates D in [brackets] for LLM models. For LLMs, we chose a test sample of 50 graphs. * indicates the lowest denial rate for each modality. The highest accuracy rate for the dataset is in bold, while the second highest is underlined.

training and test set details are given in Table 12. In the main paper, we report the GNN test accuracy on 50 samples due to API constraints by GPT-4, which do not allow us to conduct the apple-to-apple comparison with 1000 test samples. However, Table 10 reports the test accuracy for GNN reported on 1000 samples.

Dataset	Cora	Citeseer	Pubmed
Training Set	140	120	60
Testing Set	1000	1000	1000
GCN Params	23063	59366	8067
GAT Params	92373	237586	32393
GraphSage Params	46103	118710	16115

Table 12: GNN Train-Test split and Parameters