Prompt Compression via Graph Pruning

Rohit Prajapati
P24CS0201
PhD, Dept. of CSE

Introduction

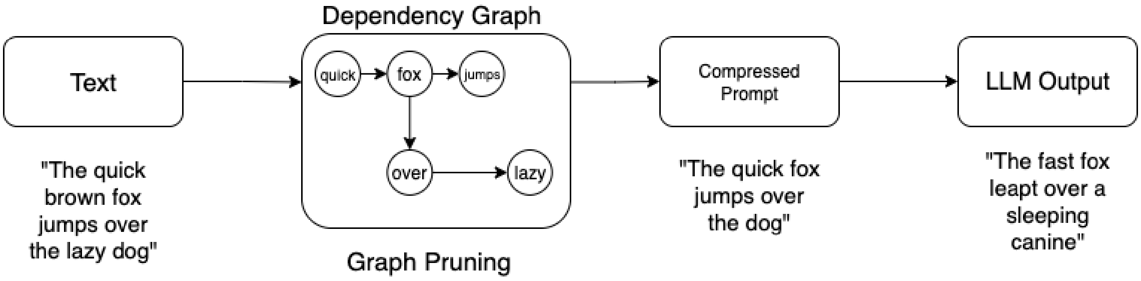
Large Language Models (LLMs) often require long input prompts, which increases both inference cost and latency. Prompt compression aims to reduce prompt length while maintaining output quality. Traditional approaches rely on heuristics such as truncation or keyword extraction.

This project proposes a graph-theoretic approach: converting prompts into dependency graphs, applying pruning techniques based on graph centrality or sparsification, and feeding the compressed prompt to an LLM. The novelty lies in leveraging graph theory for efficient prompt compression.

Objectives

The primary objective of this project is to design and evaluate a graph-based pipeline for compressing prompts in order to make LLM inference more efficient. Specifically, the project aims to reduce the number of tokens in prompts while ensuring that the compressed prompts preserve essential meaning and yield outputs of comparable quality to the original. It also seeks to systematically analyze the trade-offs between compression ratio and semantic faithfulness, thereby providing insights into how graph-theoretic techniques can optimize the balance between efficiency and performance. Another key objective is to demonstrate that this method is lightweight, reproducible, and feasible on modest hardware, while also being adaptable to larger-scale applications in industrial settings where inference costs are critical.

The Idea


Dependency Graph
Text
quick
fox
jumps
over
lazy
Compressed Prompt
LLM Output
"The quick brown fox jumps over the lazy dog"
Graph Pruning
"The quick fox jumps over the dog"
"The fast fox leapt over a sleeping canine"

Databases and Evaluation Metrics

Datasets

The experiments will be conducted using well-established summarization datasets such as CNN/DailyMail and XSum, which are widely used benchmarks for text generation tasks. Additionally, a small custom dataset of research-related prompts may be prepared to evaluate the approach in more domain-specific contexts.

Evaluation Metrics

The effectiveness of the method will be assessed using multiple metrics. The **compression ratio** will quantify token savings relative to the original prompt. **ROUGE** and **BLEU** scores will measure the quality of generated outputs, while semantic similarity will be evaluated using cosine similarity between embeddings (e.g., Sentence-BERT). Together, these metrics will capture both efficiency and fidelity.

Methodology

The methodology begins with the construction of dependency graphs from input prompts using a parser such as spaCy or Stanford CoreNLP, where nodes represent tokens and edges represent syntactic dependencies. Once the graph is built, pruning techniques are applied to reduce its size while preserving essential meaning. Centrality-based pruning identifies and retains tokens of high importance (e.g., based on degree or PageRank), while low-importance tokens are removed. Spectral sparsification can also be employed to approximate the graph with fewer edges while maintaining its structural integrity. After pruning, the remaining graph is linearized back into text to form the compressed prompt. Both the original and compressed prompts are then fed into an LLM, and their outputs are compared using quality and semantic similarity metrics.

Expected Outcomes

The project is expected to show that graph pruning can significantly reduce prompt length without a corresponding drop in generation quality. By comparing the LLM outputs from compressed and original prompts, we aim to demonstrate a clear trade-off between compression ratio and semantic fidelity. The framework should prove lightweight and reproducible, requiring minimal hardware resources, and capable of delivering measurable improvements in efficiency. Beyond academic interest, the results are expected to have industrial relevance by reducing the token usage and inference costs of LLMs in real-world applications.

References

[1] Prompt compression with context-aware sentence selection. In *AAAI*, 2025.

[2] Anonymous. Prune-on-logic: Can pruning improve reasoning? revisiting long-cot. *arXiv preprint arXiv:2505.14582*, 2025.

[3] Lizhe Chen, Binjia Zhou, Yuyao Ge, et al. Pis: Linking importance sampling and attention mechanisms for efficient prompt compression. *arXiv preprint arXiv:2504.16574*, 2025.

[4] Gongfan Fang, Xinyin Ma, Mingli Song, Michael Bi, and Xinchao Wang. Depgraph: Towards any structural pruning. In *CVPR*, 2023.

[5] Weizhi Fei, Xueyan Niu, et al. Efficient prompt compression with evaluator heads for long-context transformer inference. *arXiv preprint arXiv:2501.12959*, 2025.

[6] Song Guo, Jiahang Xu, Li Lyna Zhang, and Mao Yang. Compresso: Structured pruning with collaborative prompting learns compact llms. *arXiv preprint arXiv:2310.05015*, 2023.

[7] Torsten Hoefler, Dan Alistarh, Tal Ben-Nun, Nikoli Dryden, and Alexandra Peste. Sparsity in deep learning: Pruning and growth for efficient inference and training. *Journal of Machine Learning Research*, 2021.

[8] Yixuan Jiang, Hui Pan, et al. Characterizing prompt compression methods for long contexts: Llmlingua, longllmlingua, selective-context, and pcrl. *arXiv preprint arXiv:2407.08892*, 2024.

[9] Su Li and Nigel Collier. 500xcompressor: Generalized prompt compression for large language models. In *ACL*, 2025.

[10] Hongwu Peng et al. Towards sparsification of graph neural networks. *arXiv preprint arXiv:2209.04766*, 2022.

[11] Xiao Pu, Tianxing He, and Xiaojun Wan. Style-compress: An llm-based prompt compression framework considering task-specific styles. In *Findings of EMNLP*, 2024.

[12] Shujian Yu, Francesco Alesiani, Wenzhe Yin, Robert Jenssen Jenssen, and Jose C. Principe. Principle of relevant information for graph sparsification. *Proceedings of Machine Learning Research*, 2022.

[13] Tinghui Zhang, Yifan Wang, and Daisy Zhe Wang. Scope: A generative approach for llm prompt compression. *arXiv preprint arXiv:2508.15813*, 2025.

[14] Ye Zhang et al. Graph convolution over pruned dependency trees improves relation extraction. In *ACL*, 2018.

[15] Zhang Zheng, Jinyi Li, et al. An empirical study on prompt compression for large language models. *arXiv preprint arXiv:2505.00019*, 2025.