Decomposition Does Not Help: Evidence from Semantic Clustering in LLM-based Causal Graph Discovery

Nikolay Babakov, Alberto Bugarín-Diz
CiTIUS-Centro Singular de Investigación
en Tecnoloxías Intelixentes
Universidade de Santiago de Compostela
{nikolay.babakov, alberto.bugarin.diz}@usc.es

Abstract

Recent advances in large language models (LLMs) have enabled their application to non-traditional tasks such as causal graph construction, a key component of reasoning frameworks, including Bayesian Networks. The most effective existing approaches rely on direct prompting, where an LLM generates a complete graph from a full set of variables in a single step. However, the performance of such methods degrades as the number of graph nodes increases. To address this limitation, we explore a divide-and-conquer alternative based on semantic clustering. Node representations are first embedded and clustered, after which subgraphs are constructed independently for each cluster using LLM prompting. The resulting subgraphs are then merged pairwise into a global graph.

Contrary to our expectations, this approach leads to a substantial degradation in performance compared to direct prompting baselines, as measured by Structural Hamming Distance (SHD). We attribute this to the misalignment between semantic similarity and causal structure, as well as error propagation during subgraph merging. We report these negative results to highlight the limitations of decomposition strategies in LLM-based causal graphs construction.

1 Introduction

The growing capabilities of large language models (LLMs) have expanded their applications into domains not traditionally associated with natural language processing, including education (Kasneci et al., 2023) and programming (Guo et al., 2024). One such emerging application is causal graph (CG) construction, a key component of probabilistic reasoning frameworks like Bayesian Networks (BNs) (Koller, 2009). Causal graph discovery (CGD) has traditionally been addressed either through data-driven structure learning algorithms (Kitson et al., 2023) or through expert elicitation, where domain knowledge is used to define causal relationships (Nyberg et al., 2022) . In contrast to these approaches, recent work demonstrates that LLMs can infer causal structure directly from textual descriptions of variables, effectively enabling CGD without explicit data or expert input (Wang et al., 2024; Chen et al., 2024; Wan et al., 2024).

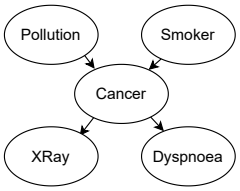

Figure 1: Causal Graph of the BN related to the lung cancer problem (Korb and Nicholson, 2010).

CG is a directed acyclic graph (DAG) that illustrates variables and their causal dependencies. Consider the example shown in Figure 1, which depicts a CG of a simple BN (Korb and Nicholson, 2010). This BN models a hypothetical scenario involving potential causes (e.g., Pollution and Smoker) and effects (e.g., X-Ray results and Dyspnoea) of Lung Cancer.

The growing attention to LLM-based causal graph construction has led to the development of dedicated evaluation benchmarks, such as CausalGraphBench introduced by Babakov et al. (2025b), enabling systematic comparisons of different approaches. In addition to establishing a unified evaluation setting, this work demonstrates that simple direct prompting strategies, where an LLM is asked to reconstruct a full graph from a list of nodes in a single step, perform on par with more elaborate multi-step methods that incorporate additional reasoning or constraints. At the same time, all evaluated approaches exhibit a substantial decline in

Proceedings of the 1st Symposium on Natural Language Generation Evaluations, pages 1–7

performance as graph size increases, identifying scalability as a central challenge in LLM-based CGD (Babakov et al., 2025b).

This limitation motivates the exploration of decomposition strategies that break the task into smaller, more manageable subproblems. In this work, a divide-and-conquer approach based on semantic node clustering is investigated. Given textual descriptions of variables, nodes are embedded into a vector space, dimensionality is reduced, and clusters are formed using a pipeline inspired by topic modelling techniques. Each cluster is then processed independently by an LLM to construct a subgraph, after which the resulting subgraphs are iteratively merged in a pairwise manner to produce a global causal graph.

The approach is evaluated using the aforementioned CausalGraphBench benchmark. Contrary to expectations, decomposition via semantic clustering results in substantial performance degradation compared to direct prompting baselines. These results suggest that semantic similarity between node descriptions does not align well with underlying causal structure and that errors introduced during subgraph construction and merging accumulate in the final graph.

By reporting these negative findings, this work aims to contribute to a better understanding of the limitations of decomposition strategies in LLM-based structured prediction tasks and to inform future research on scalable approaches to causal graph construction.

2 Related works

LLMs have been explored for a variety of graph-related tasks, including connectivity, cycle detection, shortest path, and topological ordering (Wang et al., 2024; Chen et al., 2024).

In the context of causal graph construction, existing approaches can be broadly divided into two categories. The first combines LLMs with traditional data-driven methods (Ban et al., 2023a; Long et al., 2023a). The second category relies on LLMs to construct causal graphs directly, with methods differing mainly in how they query the model. One group of LLM-only methods makes exhaustive queries, like all possible pairs, triplets, or other combinations of nodes, resulting in a significant number of queries necessary for reconstruction of one CG (Cohrs et al., 2024; Zhang et al., 2024; Vashishtha et al., 2023; Long et al., 2023b; Kıcıman et al., 2023; Feng et al., 2024; Darvariu et al., 2024; Zhou et al., 2024). In contrast, minimal-query approaches aim to construct the full graph with fewer interactions while preserving a more global view of the structure, including iterative graph construction, structured multi-step prompting, and ensemble-style aggregation of independently generated graphs (Jiralerspong et al., 2024; Ban et al., 2023b; Babakov et al., 2025a; Zhang et al., 2025).

3 Experimental setup

3.1 Dataset

The experiments are conducted using the CausalGraphBench benchmark (Babakov et al., 2025b), which comprises 35 causal graphs derived from both publicly available repositories and academic papers. The benchmark includes graphs of varying sizes, with a median of 16 nodes and 21 edges. Each graph is accompanied by structured metadata, including a textual description of the graph's purpose, the associated knowledge domain, a dictionary of node descriptions clarifying variable semantics, and the ground-truth graph structure.

3.2 Methodology of the experiments

Two approaches to CG construction are compared: a baseline method and a cluster-based decomposition method.

The baseline follows a direct zero-shot prompting strategy, where the LLM is provided with the full list of clearly defined node names and asked to generate the complete causal graph in a single step.

The cluster-based method introduces a multi-step pipeline to decompose the task into smaller subproblems, motivated by scalability challenges observed in prior work. The approach follows a subset of steps inspired by the BERTopic framework (Grootendorst, 2022). First, node descriptions are embedded into a vector space using sentence embedding models; two variants are considered: MiniLM[1] and Gemma[2]. Second, dimensionality reduction is applied using UMAP (McInnes et al., 2018; McInnes et al., 2018). Third, clustering is performed using HDBSCAN (McInnes et al., 2017). The hyperparameters for these stages are adopted based on recommendations from BERTopic: minimum cluster size of 2, UMAP with

[1] huggingface.co/sentence-transformers/all-MiniLM-L6-v2
[2] huggingface.co/google/embeddinggemma-300m

15 neighbours (capped at the number of nodes minus one), 2 output components, cosine distance metric, and HDBSCAN with Euclidean distance and 'excess of mass' cluster selection.

After clustering, each cluster is processed independently by the LLM using the same prompting strategy as in the baseline, producing subgraphs. Clusters containing a single node are preserved without modification. The resulting subgraphs are then combined through a pairwise merging procedure: all pairs of subgraphs (or a restricted subset based on nearest cluster centroids) are presented to the LLM, which is queried to infer cross-cluster connections. The final causal graph is constructed by aggregating intra-cluster subgraphs and inter-cluster edges. Two merging strategies are explored: considering all possible subgraph pairs and restricting merging to the top-k nearest clusters (with $k \in 3, 5$) based on cosine distance between cluster centroids. The specific prompts used for LLM querying are shown in the Appendix.

Experiments are conducted using both proprietary and open-source LLMs. The proprietary models include GPT-5.4 (2026-03-05)[3] and GPT-5.2 (2025-12-11)[4]. Open-source models include GPT-OSS-120b (OpenAI, 2025) and GLM-5 (GLM-5-Team et al., 2026).

3.3 Evaluation

We evaluate the quality of the LLM-generated CGs using Structural Hamming Distance (SHD), a widely used measure for evaluating graph discovery algorithms (Tsamardinos et al., 2006). Lower SHD values indicate higher-quality graphs. SHD is calculated as the total number of operations (addition, removal, or reversal of edge directions) required to transform the generated graph into the target graph. Incorrectly oriented edges, where the cause and effect are reversed, are penalised as two errors. To make comparisons across CGs of varying sizes more meaningful, we report SHD normalised by the node count in the actual CG. We used causal discovery toolbox[5] for SHD calculations.

4 Contamination Analysis

LLMs may have prior exposure to some CGs, leading to artificially improved performance (Tu et al., 2023; Tamkin et al., 2021; Sainz et al., 2023). To mitigate this, a contamination detection procedure based on Babakov et al. (2025a) is applied. Each model is first prompted to reconstruct the set of nodes of a CG using only its metadata (paper and, when available, source URL). Exact recovery of nodes in both number and semantic meaning is treated as a signal of potential contamination.

CG name	LLM	SHD/nodes
alarm	GPT-5.4	0.41
cancer	GLM-5	0
	GPT-5.2	0.2
	GPT-5.4	0
	GPT-OSS-120b	0
coma	GPT-5.4	0
covid	GPT-5.4	0.2
sachs	GLM-5	0.73
	GPT-5.4	0.73

Table 1: Results of the second step of contamination analysis - for CGs that are potentially contaminated (i.e., LLM can produce an accurate list of nodes relying solely on paper name or URL), LLM is also queried to generate a corresponding CG.

Using this procedure, such signals are observed for several models, including GLM-5 (*sachs*, *cancer*), GPT-5.2 (*cancer*), GPT-5.4 (*sachs*, *cancer*, *alarm*, *covid*, *coma*), and GPT-OSS-120b (*cancer*). These CGs are further tested by prompting the corresponding models to reconstruct their structure from the generated nodes. The resulting graphs are compared to the ground truth using normalized SHD (Table 1).

The results show that, although multiple CGs have perfectly reconstructed node sets, only a subset can be accurately recovered at the structural level. In particular, *cancer* and *coma* are reconstructed with a zero error by at least one model, indicating strong prior exposure. These two CGs are therefore excluded from further experiments to ensure fair evaluation.

5 Experimental Results

The experiments are conducted by applying both the baseline and the cluster-based methods to all CGs that were not excluded during the contamination analysis (Section 4). The cluster-based method is evaluated with two different embedding models (Section 3.2).

Table 2 presents the results for the all-vs-all merging strategy, where the LLM is queried to merge every possible pair of subgraphs. The results show a substantial increase in SHD compared to

[3] openai.com/index/introducing-gpt-5-4/
[4] openai.com/index/introducing-gpt-5-2/
[5] github.com/ElementAI/causal_discovery_toolbox

Mehtod/LLM	GPT-5.4	GPT-5.2	GPT-OSS-120b	GLM-5
Baseline	1.71	1.67	1.68	1.57
Cluster (MiniLM)	3.04	3.52	3.26	3.09
Cluster (Gemma)	2.74	3.10	3.00	2.81

Table 2: SHD normalized by nodes count for baseline experiments and cluster-based experiments, conditioned on clusters from MiniLM and Gemma encoder models.

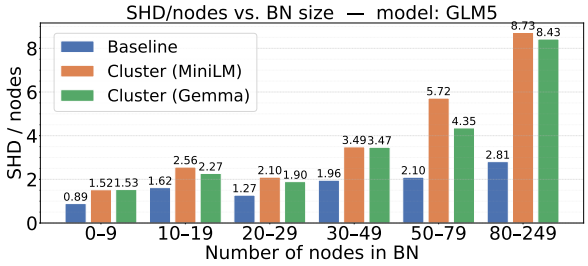

Figure 2: Normalised SHD for CGs of different sizes resulting from application of baslein and cluster-based method with GLM-5 model.

the baseline across all models, indicating that this approach is ineffective for CGD. This trend is further illustrated in Figure 2, where the degradation in performance is observed even for small graphs. As the number of nodes increases, the negative effect becomes significantly more pronounced, highlighting the poor scalability of the cluster-based decomposition under exhaustive merging.

An alternative merging strategy, in which only the top-3 or top-5 nearest cluster pairs (based on centroid proximity) are considered instead of all possible pairs, also fails to yield meaningful improvements. As shown in Appendix Table 3, restricting the merging process does not significantly reduce SHD, indicating that limiting inter-cluster interactions is insufficient to overcome the limitations of the cluster-based approach.

6 Discussion

The contamination analysis reveals that, although several widely known CGs have node sets that are clearly recognised by the models, this knowledge does not reliably translate into accurate reconstruction of the underlying graph structure. Even when node names are perfectly recovered, the corresponding causal relationships are often not. This suggests that causal graph reconstruction is not a well-internalised capability of LLMs. While some models (e.g., GPT-5.x) support multimodal inputs, this does not imply effective retention or use of structured graph knowledge. These findings support the validity of the CausalGraphBench benchmark, as the task does not reduce to memorisation, and highlight the importance of contamination checks for reliable evaluation.

The main experimental results demonstrate that the proposed clustering-based decomposition does not improve causal graph construction and, in fact, leads to substantial performance degradation. While the approach was intended to simplify the task and improve scalability for larger graphs, the opposite effect is observed: errors increase significantly as graph size grows. A likely explanation is that semantic clustering of node descriptions does not correspond to the underlying causal structure. As a result, important cross-cluster dependencies are lost, and the subsequent merging process introduces additional inconsistencies, ultimately leading to higher reconstruction error compared to direct prompting.

7 Conclusion

This work investigates a clustering-based decomposition strategy for LLM-driven causal graph construction and finds that, contrary to expectations, it consistently degrades performance compared to direct prompting. The results show that semantic node grouping does not align with the causal structure, and decomposition introduces errors that accumulate during graph merging, particularly in larger graphs. These findings highlight the limitations of naïve divide-and-conquer approaches for structured reasoning with LLMs and suggest that preserving global context is critical for accurate causal graph discovery.

Acknowledgments

The first author would like to express sincere gratitude to Ehud Reiter for his guidance and co-supervision during the PhD studies at the University of Santiago de Compostela. This work is a direct continuation of prior joint research on the application of LLMs to causal graph construction, and several ideas explored in this paper, including

the contamination analysis procedure, originated from discussions within that collaboration.

This paper is part of the R+D+i projects PID2023-149549NB-I00 and PID2023-149959OA-I00 funded by MCIN/AEI/10.13039/501100011033/ and by "ERDF a way of making Europe". The support of the Galician Ministry for Education, Universities and Professional Training and "ERDF A way of making Europe" is also acknowledged through the grant "Centro de investigación de Galicia accreditation 2024-2027 ED431G-2023/04".

References

Nikolay Babakov, Ehud Reiter, and Alberto Bugarín. 2025a. Scalability of Bayesian network structure elicitation with large language models: a novel methodology and comparative analysis. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10685–10711, Abu Dhabi, UAE. Association for Computational Linguistics.

Nikolay Babakov, Ehud Reiter, and Alberto Bugarín-Diz. 2025b. CausalGraphBench: a benchmark for evaluating language models capabilities of causal graph discovery. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 240–258, Vienna, Austria. Association for Computational Linguistics.

Taiyu Ban, Lyuzhou Chen, Derui Lyu, Xiangyu Wang, and Huanhuan Chen. 2023a. Causal structure learning supervised by Large Language Model. *arXiv preprint arXiv:2311.11689*.

Taiyu Ban, Lyvzhou Chen, Xiangyu Wang, and Huanhuan Chen. 2023b. From query tools to causal architects: Harnessing Large Language Models for advanced causal discovery from data. *arXiv preprint arXiv:2306.16902*.

Sirui Chen, Mengying Xu, Kun Wang, Xingyu Zeng, Rui Zhao, Shengjie Zhao, and Chaochao Lu. 2024. CLEAR: Can language models really understand causal graphs? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 6247–6265, Miami, Florida, USA.

Kai-Hendrik Cohrs, Gherardo Varando, Emiliano Diaz, Vasileios Sitokonstantinou, and Gustau Camps-Valls. 2024. Large Language Models for constrained-based causal discovery. *arXiv preprint arXiv:2406.07378*.

Victor-Alexandru Darvariu, Stephen Hailes, and Mirco Musolesi. 2024. Large Language Models are effective priors for causal graph discovery. *arXiv preprint arXiv:2405.13551*.

Tao Feng, Lizhen Qu, Niket Tandon, Zhuang Li, Xiaoxi Kang, and Gholamreza Haffari. 2024. From pretraining corpora to Large Language Models: What factors influence LLM performance in causal discovery tasks? *arXiv preprint arXiv:2407.19638*.

GLM-5-Team, :, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, and 168 others. 2026. Glm-5: from vibe coding to agentic engineering. *Preprint*, arXiv:2602.15763.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Yu Wu, YK Li, and 1 others. 2024. Deepseek-coder: When the Large Language Model meets programming–the rise of code intelligence. *arXiv preprint arXiv:2401.14196*.

Thomas Jiralerspong, Xiaoyin Chen, Yash More, Vedant Shah, and Yoshua Bengio. 2024. Efficient causal graph discovery using Large Language Models. *arXiv preprint arXiv:2402.01207*.

Enkelejda Kasneci, Kathrin Sessler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, Stephan Krusche, Gitta Kutyniok, Tilman Michaeli, Claudia Nerdel, Jürgen Pfeffer, Oleksandra Poquet, Michael Sailer, Albrecht Schmidt, Tina Seidel, and 4 others. 2023. ChatGPT for good? on opportunities and challenges of Large Language Models for education. *Learning and Individual Differences*, 103:102274.

Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. 2023. Causal reasoning and Large Language Models: Opening a new frontier for causality. *arXiv preprint arXiv:2305.00050*.

Neville Kenneth Kitson, Anthony C Constantinou, Zhigao Guo, Yang Liu, and Kiattikun Chobtham. 2023. A survey of Bayesian Network structure learning. *Artificial Intelligence Review*, pages 1–94.

Daphane Koller. 2009. Probabilistic graphical models: Principles and techniques.

Kevin B Korb and Ann E Nicholson. 2010. *Bayesian artificial intelligence*. CRC press.

Stephanie Long, Alexandre Piché, Valentina Zantedeschi, Tibor Schuster, and Alexandre Drouin. 2023a. Causal discovery with language models as imperfect experts. *arXiv preprint arXiv:2307.02390*.

Stephanie Long, Tibor Schuster, and Alexandre Piché. 2023b. Can Large Language Models build causal graphs? *arXiv preprint arXiv:2303.05279*.

L. McInnes, J. Healy, and J. Melville. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *ArXiv e-prints*.

Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *The Journal of Open Source Software*, 2(11):205.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. 2018. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861.

Erik P. Nyberg, Ann E. Nicholson, Kevin B. Korb, Michael Wybrow, Ingrid Zukerman, Steven Mascaro, Shreshth Thakur, Abraham Oshni Alvandi, Jeff Riley, Ross Pearson, Shane Morris, Matthieu Herrmann, A.K.M. Azad, Fergus Bolger, Ulrike Hahn, and David Lagnado. 2022. BARD: A structured technique for group elicitation of Bayesian Networks to support analytic reasoning. *Risk Analysis*, 42(6):1155–1178.

OpenAI. 2025. gpt-oss-120b and gpt-oss-20b model card. *Preprint*, arXiv:2508.10925.

Oscar Sainz, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. *arXiv preprint arXiv:2310.18018*.

Alex Tamkin, Miles Brundage, Jack Clark, and Deep Ganguli. 2021. Understanding the capabilities, limitations, and societal impact of Large Language Models. *arXiv preprint arXiv:2102.02503*.

Ioannis Tsamardinos, Laura E Brown, and Constantin F Aliferis. 2006. The max-min hill-climbing Bayesian Network structure learning algorithm. *Machine learning*, 65:31–78.

Ruibo Tu, Chao Ma, and Cheng Zhang. 2023. Causal-discovery performance of chatgpt in the context of neuropathic pain diagnosis. *arXiv preprint arXiv:2301.13819*.

Aniket Vashishtha, Abbavaram Gowtham Reddy, Abhinav Kumar, Saketh Bachu, Vineeth N Balasubramanian, and Amit Sharma. 2023. Causal inference using LLM-guided discovery. *arXiv preprint arXiv:2310.15117*.

Guangya Wan, Yuqi Wu, Mengxuan Hu, Zhixuan Chu, and Sheng Li. 2024. Bridging causal discovery and Large Language Models: A comprehensive survey of integrative approaches and future directions. *arXiv preprint arXiv:2402.11068*.

Heng Wang, Shangbin Feng, Tianxing He, Zhaoxuan Tan, Xiaochuang Han, and Yulia Tsvetkov. 2024. Can language models solve graph problems in natural language? *Advances in Neural Information Processing Systems*, 36.

Yinghuan Zhang, Yufei Zhang, Parisa Kordjamshidi, and Zijun Cui. 2025. Bayesian network structure discovery using large language models. *arXiv preprint arXiv:2511.00574*.

Yuzhe Zhang, Yipeng Zhang, Yidong Gan, Lina Yao, and Chen Wang. 2024. Causal graph discovery with retrieval-augmented generation based Large Language Models. *arXiv preprint arXiv:2402.15301*.

Yu Zhou, Xingyu Wu, Beicheng Huang, Jibin Wu, Liang Feng, and Kay Chen Tan. 2024. Causalbench: A comprehensive benchmark for causal learning capability of Large Language Models. *arXiv preprint arXiv:2404.06349*.

A Appendix

MiniLM				
Method/LLM	GPT-5.4	GPT-5.2	GPT-OSS-120b	GLM-5
Baseline (top-3)	1.95	1.90	1.87	1.80
Cluster (top-3)	2.52	2.71	2.54	2.45
Baseline (top-5)	2.03	2.00	2.02	2.02
Cluster (top-5)	3.19	3.51	3.36	3.16

Gemma				
Method/LLM	GPT-5.4	GPT-5.2	GPT-OSS-120b	GLM-5
Baseline (top-3)	1.84	1.79	1.76	1.67
Cluster (top-3)	2.16	2.25	2.37	2.17
Baseline (top-5)	1.74	1.71	1.75	1.71
Cluster (top-5)	2.39	2.67	2.66	2.54

Table 3: SHD/nodes (Structural Hamming Distance normalised by node count; lower is better) for the cluster-based approach under two inter-cluster joining strategies (top-3 and top-5 nearest-neighbour pairs) and two node encoders (MiniLM, Gemma). For each joining mode, the baseline is averaged over the same subset of CGs that satisfies the minimum cluster-count requirement for that mode, ensuring a fair comparison, e.g., for top-3 only CGs with 4 or more clusters from the corresponding encoder model are included.

A.1 Baseline prompt

This prompt was used both for a direct baseline LLM query and for querying a subgraph with a node count of more than 3.

You are an expert on {domain}. You are constructing the Bayesian Network aimed to fulfill the following task: {task}. To construct the Bayesian Network you need to investigate the cause-and-effect relationships between the following variables in your area of expertise: {variables}. Based on the meaning of variables, analyze the cause-and-effect relationships between them. Please give the results as a directed graph network. Make sure that each edge represent a direct causality between the two variables.

Return valid JSON-list of the following format: {{ "result": [[from node (A), to node(B)], # (meaning that there is a direct causal effect from node A to node B) [from node (F), to node(E))] # (meaning that there is a direct causal effect from node F to node E) [from node (D), to node(G))] # (meaning that there is a direct causal effect from node D to node G) ...] }} """

A.2 Subgraphs pairing prompt

You are an expert on {domain}. You are constructing a Bayesian Network aimed to fulfill the following task: {task}.

You are given two subgraphs of this Bayesian Network. Each subgraph is described by its nodes and the directed edges already established within it.

Subgraph 1: - Nodes: {nodes_1} - Edges: {edges_1}

Subgraph 2: - Nodes: {nodes_2} - Edges: {edges_2}

Your task is to identify direct causal relationships that exist **between** the two subgraphs — that is, edges from a node in one subgraph to a node in the other subgraph. Do NOT propose edges between nodes that are both within the same subgraph.

Return valid JSON of the following format: {{ "result": [["A", "B"], ["C", "D"]] }}

Where each pair ["A", "B"] means there is a direct causal effect from node A to node B, and A and B belong to different subgraphs.

If there are no causal relationships between the two subgraphs, return: {{ "result": [] }}