Scalability of Bayesian Network Structure Elicitation with Large Language Models: a Novel Methodology and Comparative Analysis

Nikolay Babakov
CiTIUS, Universidade de Santiago de Compostela
nikolay.babakov@usc.es

Ehud Reiter
University of Aberdeen
e.reiter@abdn.ac.uk

Alberto Bugarín
CiTIUS, Universidade de Santiago de Compostela
alberto.bugarin.diz@usc.es

Abstract

In this work, we propose a novel method for Bayesian Networks (BNs) structure elicitation that is based on the initialization of several LLMs with different experiences, independently querying them to create a structure of the BN, and further obtaining the final structure by majority voting. We compare the method with one alternative method on various widely and not widely known BNs of different sizes and study the scalability of both methods on them. We also propose an approach to check the contamination of BNs in LLM, which shows that some widely known BNs are inapplicable for testing the LLM usage for BNs structure elicitation. We also show that some BNs may be inapplicable for such experiments because their node names are indistinguishable. The experiments on the other BNs show that our method performs better than the existing method with one of the three studied LLMs; however, the performance of both methods significantly decreases with the increase in BN size.

1 Introduction

A Bayesian network (BN) is a probabilistic graphical model that represents a set of variables and their conditional dependencies through a directed acyclic graph (DAG). Refer to Figure 1 for a simple BN (Korb and Nicholson, 2010) which describes the toy scenario involving some possible reasons (Pollution and Smoker) and consequences (XRay results and Dyspnoea) of Lung Cancer.

The DAG encoding causal relationships among BN nodes is a crucial part of a BN. It can be learned from data using various BN structure learning (BNSL) algorithms (Kitson et al., 2023) or from expert knowledge (Nyberg et al., 2022). However, the recent progress of Large Language Models (LLMs) encouraged researchers to try using LLMs as another source of information to build a BN structure (Ban et al., 2023; Long et al., 2023a).

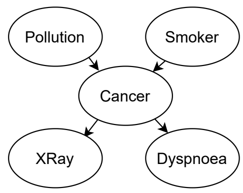

Figure 1: A BN for the lung cancer problem. Note that conditional probability tables encoding the probabilistic relations between variables are not shown here.

In this paper, we propose a new approach of LLM-driven BNSL that takes inspiration from the real-world practices of collective experts' discussion of BN structure, in particular, the Delphi protocol (Rowe et al., 1991). It assumes the presence of multiple experts who asynchronously and anonymously discuss a group of complex questions. We transfer this intuition into the LLM environment: we generate the expertise of several "LLM experts", initialize different instances of LLMs with this expertise, ask them similar questions about BN structure, and, in the end, define the finally elicited structure by majority vote.

Even though LLMs show impressive performance on various casual tasks (Tu et al., 2023), it is still necessary to understand their limitations (Tamkin et al., 2021). In terms of BNSL, the knowledge about certain BN can be obtained by an LLM during its training, which yields data contamination and results in an inadequate increase in the quality of the particular task (Sainz et al., 2023b). To the best of our knowledge, the existing works about LLM usage for BNSL run their experiments with widely-known BNs disregarding the risk of data contamination. We propose a simple prompting technique that gives a general idea of whether the information about a particular BN is known to the LLM and also involve not widely known BNs in the experiments.

Proceedings of the 31st International Conference on Computational Linguistics, pages 10685–10711
January 19–24, 2025. ©2025 Association for Computational Linguistics

The experiments performed in this work involve data contamination check and LLM-driven BNSL with our original and the existing method (Ban et al., 2023) on the BNs of different sizes (from several to hundreds of nodes) and from different sources (widely known and frequently used in BNSL papers and less known obtained from the sources less likely available to LLMs training).

The code of the developed method of LLM-driven BNSL is publicly available for easiness of reproduction.[1]

2 Related works

The structure of BN is its crucial element. If the data for a particular task is available, the structure of BN can be learned from it using structure learning algorithms (Kitson et al., 2023). Otherwise, the structure can be obtained through a collective discussion between human experts. One of the optimal ways to perform such discussion is by using the Delphi protocol (Rowe et al., 1991; Turoff and Linstone, 2002), which engages multiple experts in discussing certain problems asynchronously and anonymously under the review of the facilitator. This technique was first used for the specific task of BN structure and parameters elicitation in (Nicholson et al., 2016). Delphi protocol usage was further expanded to the whole pipeline of BN structure elicitation in the BARD system (Nyberg et al., 2022).

LLMs have shown impressive results in many tasks requiring causal analysis (Nori et al., 2023; Tu et al., 2023). The recent best practices of LLM prompting can imitate the collective mindset similar to the Delphi-style discussion. First, the LLM could be associated with a particular role on behalf of which it is asked to answer the prompt (Li et al., 2023a). Second, there already are numerous papers that engage different instances of LLMs with different "experience" (Li et al., 2023b; Wang et al., 2023b; Gao et al., 2023) or even models of different modalities (e.g., visual and language) (Zhuge et al., 2023) into discussing a target task.

However, while designing any LLM-driven method, it is necessary to take into account the phenomenon of data contamination (Sainz et al., 2023b,a). This problem is actively discussed (Jacovi et al., 2023) and numerous approaches to data contamination testing have been proposed. Some of them rely on the log-likelihood of the LLM output (Oren et al., 2023; Shi et al., 2023), on the exact data that is suspected to be contaminated (Elangovan et al., 2021; Golchin and Surdeanu, 2023b) or on special tasks (McCoy et al., 2023) and quizzes (Golchin and Surdeanu, 2023a).

Naturally, most LLM applications are directed toward fields where the information could normally be described in plain text (e.g. education (Kasneci et al., 2023)). However, the representation of graph structures in plain text is not that straightforward, but some workarounds are possible. For example, (Wang et al., 2023a) proposed specific prompting that allows using LLMs for solving general graph problems (e.g., connectivity, maximum flow). At the same time, several papers dedicated to the specific task of LLM-driven causal graph construction already exist. (Long et al., 2023b) showed that using even a very simple prompt may demonstrate proper performance of causal relations extraction. (Long et al., 2023a) use LLMs as "imperfect experts" that construct BN's structure. (Vashishtha et al., 2023) proposes four types of prompts, the most effective of which is a prompting strategy that asks LLM to elicit causal relations between all possible triplets of BN nodes. (Ban et al., 2023) proposes to use three consecutive prompts to understand the meaning of nodes, construct causal relations, and check LLM's own reasoning.

BN name	Nodes	Edges	Source
coma	5	5	bayesfusion
covid	20	26	bayesfusion
insurance	27	52	bnlearn
alarm	37	46	bnlearn
barley	48	84	bnlearn
hailfinder	56	66	bnlearn
pathfinder	109	195	bnlearn
andes	223	338	bnlearn
diabetes	413	602	bnlearn
munin	186	273	bnlearn
hepar2	70	123	bnlearn
agro	6	10	(Baudrit et al., 2022)
sperm	9	11	(Samie et al., 2022)
screen	16	21	(Zio et al., 2022)
sids	17	27	(Hamayasu et al., 2022)
apple	29	62	(Sottocornola et al., 2023)
urinary	36	107	(Ramsay et al., 2022)

Table 1: Bayesian Networks used for experiments.

[1] https://gitlab.nl4xai.eu/nikolay.babakov/delphi_lm_xpertnet

3 Experimental setup

3.1 Bayesian Networks

We aim to use BNs of different sizes and likelihoods of being seen by LLMs during training. The "popular" BNs are obtained from bayesfusion[2] and bnlearn[3] websites. The "unpopular" BNs are obtained from the papers dedicated to BN application to specific problems in various domains. See the list of all BNs engaged in our experiments in Table 1. We succeeded in finding large BNs only from the bnlearn website, because in the case of large BNs developed in the research papers, even their structure is hard to fit into the paper, and authors rarely distribute their BNs in a standalone runnable file.

3.2 Method of BN structure learning using LLMs

3.2.1 New method for BN structure elicitation

We propose a novel method for eliciting knowledge from LLM to construct a BN structure. In the proposed method, we imitate the Delphi-style discussion (experts discuss complex problems anonymously and asynchronously) for the construction of BN structure using LLMs' knowledge. Fig. 2 shows the general idea of our approach. We assume that some basic information about the BN is available: its aim and the area of necessary knowledge. First, we query facilitator LLM; its task is to generate an arbitrary odd number of experts who have unique experience relevant to the necessary knowledge area. Second, we initialize multiple LLM threads with their unique experiences and send them two consecutive prompts. The first prompt contains a list of node names and a request to discuss what kind of causal relations could be between these nodes (the node names are preliminarily clarified using *Understand* prompt discussed in Section 3.2.2). The second prompt requests LLM to summarize the reasoning from the answer to the first prompt into a JSON structure. The final BN structure is obtained by majority voting - the edge is added to the structure only if more than half of the "LLM experts" voted for it. If the cycle is found in the structure by a certain expert, it is resolved with an additional request about the conflicting pair of nodes. The full prompts for all aforementioned steps are available in Appendix A.

[2] https://repo.bayesfusion.com/bayesbox.html

[3] https://www.bnlearn.com/bnrepository/

3.2.2 Alternative method

We aim to compare our method with an alternative method that requires a sensible amount of prompts for each BN and also does not assume the presence of significant information about its structure. We disregard (Long et al., 2023b; Vashishtha et al., 2023) because the prompts proposed in these papers have to be used for each possible pair and triplet of BN nodes correspondingly, which seems impractical. Furthermore, we disregard (Long et al., 2023a), because it assumes the presence of preliminary information about BN structure obtained via some causal discovery algorithm.

Thus, for further comparative experiments, we chose the prompting strategy proposed in (Ban et al., 2023). The strategy consists of three prompts. The first prompt (*Understand*) specifies the domain area, shows the node names and the values corresponding to them, and requests LLM to understand the meaning of all nodes. The second prompt (*Causal discovery*) requests to analyze the cause-and-effect relationships between the nodes. The last prompt (*Revision*) shows LLM the structure obtained from the response to the second prompt and asks it to revise itself. All prompts and LLM answers to them are supposed to be kept in conversation history, so LLM can reuse the information elicited during previous steps. We will refer to this method as *Harness* (by the name of the paper (Ban et al., 2023)). Refer to Appendix B for the exact prompts used in this method.

3.3 Language Models

In our experiments, we use proprietary GPT-3.5 (gpt-3.5-turbo-1106) (Ouyang et al., 2022) and GPT-4 (gpt-4-0613) (OpenAI, 2023) and open-source Llama2 70b (Touvron et al., 2023).

3.4 Data contamination test

To the best of our knowledge, there is no specific test designed to evaluate the degree of contamination of BN structure for certain LLMs (i.e., how much information about a particular BN and its structure has been seen by the LLM during its training). Thus, we design a simple approach inspired by the prompt technique proposed in (Golchin and Surdeanu, 2023b). In this work, the authors employ a "guided instruction": a prompt that points to the particular piece of data to be tested for contamination and asks LLM to perform some tasks that assume that this particular piece of data has been seen by LLM during its training.

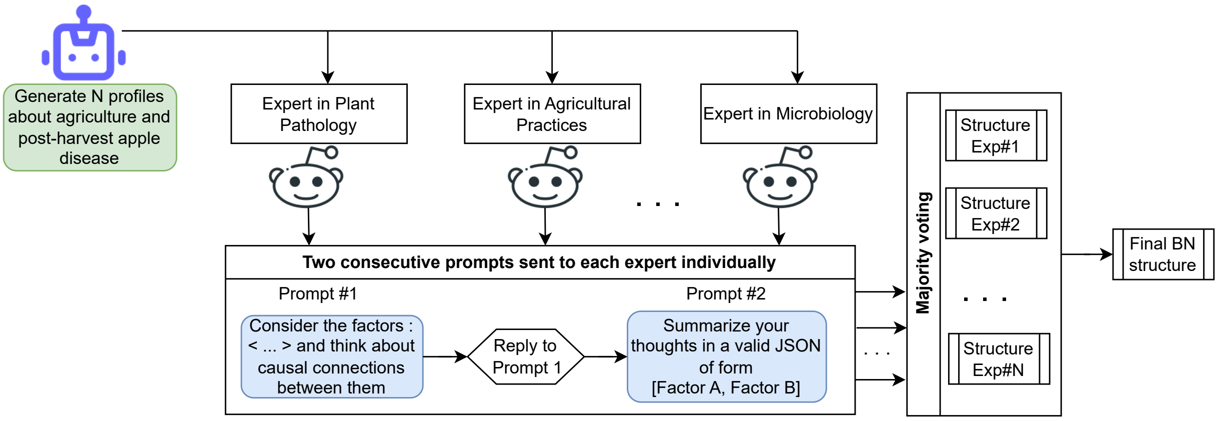

Figure 2: The overview of the proposed method for BN structure elicitation. Facilitator LLM generates N profiles relevant to the given BN. "LLM experts" initialized with different profiles are queried about the structure of BN using two consecutive prompts. The final BN structure is obtained by majority voting between the structures elicited from individual experts.

Our approach includes two prompts consecutively sent to LLM. The first prompt points to the definite paper where the BN was originally presented and to the URL address of the BN where it is publicly available (for "popular" BNs) and asks to generate the nodes of the BN. The second prompt requests to retrieve the edges that connect the elicited nodes. The full prompts are available in Appendix C.1.

3.5 Evaluation metrics

We evaluate the quality of the BN structure elicited from LLMs using the following metrics. First, we use Structural Hamming Distance (SHD) - a common metric used to evaluate graph discovery algorithms (Tsamardinos et al., 2006). Low values of SHD correspond to the better quality of the learned graph. It yields a real number that results from the sum of addition, removal, or reverse edge direction operations necessary to obtain the target graph from the learned graph. The badly oriented edges (where cause and effect are mixed) are counted as two mistakes. We report SHD normalized by real BN edge count to make the comparison of the algorithms more meaningful because the SHD increases with the increase in BN's size. We also analyze FP (false positives) indicating the number of edges to be deleted from a learned graph and FN (false negatives) indicating the number of edges to be added to a learned graph to get the structure of a real graph. Second, we retrieve the adjacency matrix (i.e. the matrix of 0 and 1, where 1 corresponds to the edge from one node to another) from elicited and target BNs' structures and calculate the F-score-macro between them.

4 Results

4.1 Ambiguous node names

Whereas the BNs easily available for reuse from such websites as bnlearn are frequently used for data-driven BNSL papers (Wang et al., 2021; Behjati and Beigy, 2020), some of these BNs seem to be impractical for the tests with LLM-driven BNSL. One important problem with such BNs is that their node names are represented as ambiguous acronyms or meaningless abbreviations. Our experiments showed that the node names of `barley` and `pathfinder` BNs are impossible to decipher using LLM (with *Understand* prompt of *Harness* approach). Thus, during deciphering multiple nodes could be understood equally by LLM. For example, in `barley` five different nodes (*nmin*, *jordn*, *ngodnt*, *ngodnn*, and *ngodn*) were deciphered as *nitrogen content in the soil used for beer ingredient production*. In the case of `pathfidner` the nodes were impossible to decipher because they are represented as abbreviature of the form *F1, F2,* .... Thus, we exclude these two BNs from further experiments.

4.2 Very large BNs

The size of the BNs is another obstacle that seems interesting for analysis. As discussed in Section 3.1 we engage several large BNs from the bnlearn website to check how the studied LLM-driven BNSL methods perform on them. The largest BNs (`andes` with 223 nodes, and `diabetes` with 413 nodes)

BN	True#	Llama2		GPT-3.5		GPT-4	
		#	Rec	#	Rec	#	Rec
coma	5	35	0.2	7	0	0	0
covid	20	27	0.05	9	0.1	0	0
insurance	27	9	0.04	7	0	17	0.52*
alarm	37	15	0.03	10	0.24*	37	0.86*
hailfinder	56	14	0.11	7	0	20	0.04
hepar2	70	20	0.19	12	0.13	15	0.16
agro	6	14	0	10	0.17	0	0
sperm	9	14	0	9	0	0	0
screen	16	14	0.06	7	0.06	0	0
sids	17	5	0.18	8	0.24	0	0
apple	29	14	0	5	0	0	0
urinary	36	14	0.03	8	0.06	0	0

Table 2: The results of data contamination experiments. # corresponds to the number of generated nodes, and Rec corresponds to a recall of true nodes relative to generated nodes. The cases where the LLM used the node names that were precisely equal to the ones used in the publicly available BN are marked with a "*" sign.

exceeded the requirements of the context size available to most of the LLMs engaged in the experiments. *Understand* prompt for andes BN did not fit into the GPT-4 and Llama 2 contexts. Whereas it fit into the GPT-3.5 context (recall, that we used gpt-3.5-turbo-1106 which had a bigger context size than gpt-4-0613), the LLM failed to decipher the meaning of all nodes. *Understand* prompt for diabetes did not fit into the context size of any of the LLMs. *Understand* prompt for munin BN (186 nodes) fit into the context size of all LLMs, but similarly to andes none of the LLMs succeeded in the deciphering of all nodes of this BN. The prompts required for running our method with the aforementioned BNs also did not fit into the context size of all LLMs. Thus, we exclude these BNs from further experiments as well.

4.3 Data contamination

We check the contamination of the information about each BN for each engaged LLM as discussed in Section 3.4 and analyze the results manually. We first check the list of nodes generated by the LLMs and map them to the node names of real BN if the sense of the generated node is equal to the real node. We analyze the obtained results by comparing the true number of the nodes in a target BN with the number of nodes generated by LLM, and also by calculating the recall of the target nodes (the split of real nodes that were generated by LLM).

The results of the experiments (Table 2) show that the information about any BN does not seem to be known to Llama2. However, it is very likely that insurance BN is known to GPT-4 and alarm BN is known to both GPT-3.5 and GPT-4. In the case of GPT-3.5, even though the recall of BN nodes is not very high, the way the LLM referred to the node names in its answer is precisely similar to the names used in its publicly available version on bnlearn. In the case of GPT-4, the recall for both of these BNs is very high (compared to all other cases), the LLM also uses the node names similar to the ones in publicly available BN, and in the case of alarm the LLM even generates the precise number of nodes included in a target BN.

LLM	BN	True #	% edg	F-score	SHD
GPT-3.5	alarm	46	0.17	0.51	51
GPT-4	alarm	46	0.91	0.63	60
GPT-4	insurance	52	0.46	0.64	52

Table 3: The analysis of the BN structure elicited from LLM knowledge in terms of data contamination test. "True #" indicates the true number of edges in the BN. "% edg" indicates the ratio of the number of generated edges to the true number.

Overall, it is highly probable that Llama2 and GPT-3.5 (excluding the alarm BN), derive their information from the name of the paper mentioned in the prompt and attempt to predict the nodes that a BN described in the mentioned paper might include. We verify this idea by prompting LLMs with the names of two non-existing papers. Llama2 and GPT-3.5 keep generating the node names similarly to what they did for the real BNs. At the same time, GPT-4 seems to be more careful in its answers. As shown in Table 2 GPT-4 has a lot of cases where no nodes were generated. In these cases, the general sense of GPT-4's reply was that the corresponding paper does not provide a detailed list of nodes BNs. In some cases (e.g., coma, covid) it generated a list of nodes that could be included in such a BN (similarly to other BNs) but it was specifically indicated in the reply that these nodes are elicited from the idea of the paper name. In such cases, we assume that no nodes have been provided. Refer to Appendix C.2 for examples of contamination test results for some BNs.

The contamination tests with other BNs yield very low recall, and the number of nodes generated by LLMs significantly differs from the true number. The cases when the recall is around 0.2 (hepar2 and sids) are most likely caused by the explicit names of the papers and the awareness of the LLMs of the concepts of these papers (hepar2 - diagnosis of liver disorders, sids - sudden unexpected infant death).

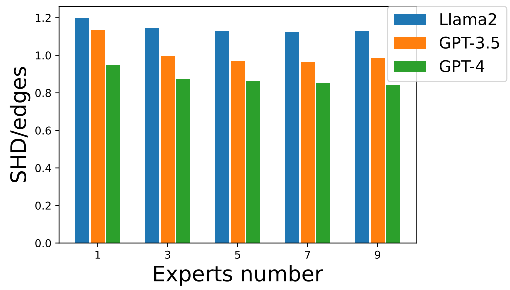

Figure 3: Dynamic of mean SHD normalized by real edges count related to the number of “LLM expert” profiles.

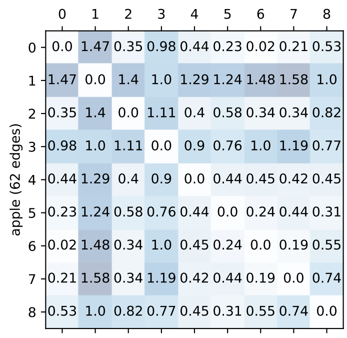

Figure 4: SHD normalized by real edges count between structures generated by different GPT-4 “LLM-experts” for `apple` BNs.

We also analyze the quality of the structure elicited by the LLMs in response to the second prompt of the contamination test. Such analysis makes sense only in cases where the response to the prompt about node names gives reason to suspect that the data are contaminated. Thus, we analyze the elicited structure only for three cases: `alarm` for GPT-3.5 and GPT-4 and *insurance* for GPT-4 (refer to Table 3). In the case of GPT-4 for both BNs, a significant number of edges were generated, resulting in an F-score higher than 0.6, which increases the suspicions that the BN structure is compromised. In the case of GPT-3.5, the elicited structure does not seem to be sensible.

We aim to perform further experiments only on the BNs that are unseen by all LLMs, so we exclude `alarm` and `insurance` BNs, and based on contamination test results, we assume that the rest of the BNs have not been seen by all LLMs engaged in our study. Refer to Appendix D Figure 6 for the visualization of all the steps for filtering the BNs to be used for the experiments with LLM-driven BNSL.

4.4 Multiple “LLM experts”

The setup of our method of LLM usage for BNSL has a hyperparameter that needs to be studied and fixed for further experiments - the number of experts.

To study the effect of the number of expert profiles generated by the facilitator, we launch our method on all BNs that have not been previously excluded from the experiments with odd numbers of experts from 1 to 9 and calculate the mean SHD from all BNs with the corresponding expert profiles. Figure 3 shows that there is a positive effect of multiple expert profiles (recall that lower SHD corresponds to the better quality of the learned graph), i.e., three or more experts perform better than a single one. However, there is no best number of experts. For uniformity of further experiments, we fixed the number of experts to 7, because this seems optimal to engage a significant number of different “opinions” from different “LLM experts” and, at the same time, it gives multiple results for every BN (i.e., 36 combinations of experts), so we may check the robustness of the proposed method w.r.t. different “LLM expert” profiles.

BN	# edges	Llama2	GPT-3.5	GPT-4
coma	5	0.67	0.61	0.73
agro	10	0.24	0.44	0.5
sperm	11	0.57	0.8	0.09
screen	21	0.4	0.78	0.4
covid	26	0.81	0.47	0.22
sids	27	0.47	0.42	0.42
apple	62	0.29	0.55	0.63
hailfinder	66	0.1	0.54	0.29
urinary	107	0.42	0.37	0.2
hepar2	123	0.51	1.01	0.18
mean		*0.45*	*0.6*	*0.37*

Table 4: Mean SHD normalized by real edges number between the BN structures proposed by 9 “LLM-experts”.

We also check how the structures proposed by different instances of LLMs differ from each other. Figure 4 shows that BN structures generated by 9 “LLM-experts” for `apple` BN have significant differences. Table 4 reporting the mean SHD normalized by edge count of each confusion matrix of this type obtained for all LLMs and BNs shows that the structures proposed by different LLM instances differ from each other and the difference varies: GPT-3.5 generates the most diverse BN structures, whereas multiple GPT-4 “LLM experts” tend to generate more uniform structures. See the detailed overview of inter-experts SHD for all LLMs and BNs in Appendix D Figure 7 and 8.

BN	#edg	Llama2		GPT-3.5		GPT-4	
		harn	our	harn	our	harn	our
coma	5	1.6	1	0.4	0.6	0.4	0
agro	10	0.8	1.1	1.1	1.2	1.1	1
sperm	11	1.27	0.91	1.18	1.09	1	0.73
screen	21	1.24	1.33	1.48	1.1	1.29	1.05
covid	26	0.65	1.15	0.69	0.85	0.58	0.58
sids	27	1	1.04	1.44	0.96	1.15	1.22
apple	62	1.02	1.37	1.15	0.85	0.85	0.94
hailfinder	66	1.05	1.33	1.24	1.05	1.03	1.03
urinary	107	1.01	1.01	1.97	0.94	0.99	1.01
hepar2	123	1.07	1.13	1.17	1.06	1.04	1
mean		1.07	1.14	1.18	0.97	0.94	0.86

Table 5: SHD normalized by real edges count of *Harness* and our method.

4.5 Comparison of the methods

Table 5 shows the SHD normalized by the number of edges resulting from the two methods. We apply the Wilcoxon signed-rank test, which shows, that there is no significant difference in the performance of the methods with Llama2 and GPT-4 (p-values of 0.43 and 0.17, respectively), whereas our method performs significantly better than *Harness* with GPT-3.5 (p-value 0.04).

We show more detailed statistics of both methods with GPT-4 in Table 6 (similar statistics for GPT-3.5 and Llama2 can be found in Appendix D Table 7), where we see that the false negatives (i.e., the necessary edges that have not been included in the final graph) make the greatest contribution to SHD for both methods. It seems, that even though the sensible causal connections could be elicited by the best-performing GPT-4, it still cannot cover them all, especially for big BNs. Moreover, the F-score has a drastic decrease when the size of BN increases - experiments with `sids` (17 nodes, 27 edges) and bigger BNs result in a pretty low F-score, which suggests that the overall quality of the results for BNs of such size drops significantly. This could also be seen in Figure 5, where we show the dynamics of SHD and F-score by the number of edges, and Appendix D Figure 9 by nodes. All combinations of methods and LLMs do not seem capable of generating the BN structure with SHD significantly less than the number of edges. The F-score begins to border on random guessing for all BNs with more than 60 edges for all combinations of LLMs and methods as well.

5 Error analysis

In this section, we discuss several errors found typical for different LLMs applied for both studied methods. The examples of all errors can be found

BN	Method	FP	FN	F-score	SHD/edg
coma	har	0	2	0.85	0.4
	our	0	**0**	**1**	**0**
agro	har	**3**	8	0.54	1.1
	our	4	**6**	**0.62**	**0.98**
sperm	har	3	8	0.64	1
	our	3	**5**	**0.77**	**0.73**
screen	har	9	18	0.56	1.29
	our	**4**	18	**0.59**	**1.05**
covid	har	**0**	15	0.79	0.58
	our	2	**12**	**0.82**	0.57
sids	har	**6**	25	0.53	**1.15**
	our	8	**24**	**0.54**	1.21
apple	har	**5**	**48**	**0.66**	**0.85**
	our	8	49	0.63	0.93
hailfinder	har	6	**62**	**0.55**	1.03
	our	**2**	66	0.5	1.03
urinary	har	**5**	**101**	**0.53**	**0.99**
	our	6	102	0.52	1.01
hepar2	har	7	121	0.51	1.04
	our	**5**	**118**	**0.53**	**1**

Table 6: Detailed information about the performance of *Harness* and our method with GPT-4. FP and FN correspond to false positive and false negative edges in the learned BN structure.

in the Appendix E.

Some LLMs, in particular, GPT-3.5 are prone to generating cycles (i.e., two edges of the graph directed in different directions between *node A* and *node B*). It turns out to be most critical for our method, whereas *Harness* method generates almost no cycles with all LLMs. See the detailed statistics of cycles in Appendix E Tables 8 and 9.

The ability to stick to the node names proposed for creating the BN structure is another problem that yields errors in our method. Recall that we use the maximally unambiguous names of the nodes, explained with *Understand* prompt of *Harness* method. On the one hand, it gives the LLM a clear input, but on the other hand, when the LLM generates a final JSON, it may fail to accurately reproduce all the names of the nodes. Sometimes such errors are related to such nonsensible differences as capitalization changes. In such cases, we try to find the most similar name of the node using Levenshtein distance. If no matches are found, we have to manually analyze the output and decide what to do with the nodes, whose names are significantly different from any nodes provided to create a BN structure. It turns out that LLMs may "invent" the nodes. Sometimes, it may take the node name either from the system message or its reasoning (answer to the first prompt), where new terminology appears. For example, Llama2 instantiated as *Epidemiologist with 15 years of experience in studying*

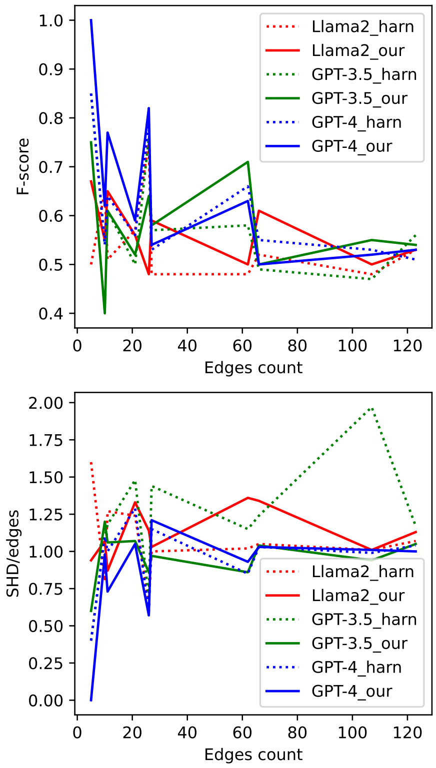

Figure 5: F-score and SHD normalized by edges count related to the number of edges in a BN.

*the distribution and determinants of health-related events, diseases, or health-related characteristics among populations, including the epidemiology of **urinary tract infections** in children* generated a BN structure of `urinary` BN with most edges connected to a node *Urinary tract infection*, which was not proposed as a possible node name in the prompts.

Sometimes LLM may refuse to generate any structure because of ethical or some other reasons (e.g. *"As a medical ethicist, I am not qualified to determine the direct relationships between these medical factors."*). In such cases, we interpret the resulting JSON as an empty list.

Finally, in some cases, LLM seems to misunderstand the idea of causal relations typical for BN structure, and instead of complex analysis, it may start generating a cause-and-effect chain.

6 Discussion

In this section, we discuss the main insights of our study.

First, the proper results of LLM usage for BNSL require proper input. Whereas in the case of data-driven algorithms, the meaning of nodes is not important, for LLM-driven BNSL, the maximally disambiguated sense of each node is crucial because it is to be used for further reasoning about causal relationships between the nodes. There is no way to control the clarity of BN nodes when we test the algorithm on the existing BNs. For using any of the discussed algorithms for the real task, the clarity of the nodes turns out to be crucial. Moreover, in the case of using our approach, we experimented with LLM-generated expert profiles, which is another point to be controlled. The most optimal way could be to have a domain specialist capable of unambiguous definitions of all input values for launching the BN structure elicitation: node names, knowledge area, and profiles of "LLM experts".

Second, it is clear that the performance of both of the discussed methods significantly decreases when the number of nodes and edges in BNs increases. In some cases, the problems were caused by the context size of the LLM selected for the experiments. This could be solved with LLMs with bigger context (e.g. gpt-4-32k), but the experiments show that the increase of LLM context size is unlikely to solve the main problem, which seems to be the inability of all LLMs engaged in the experiments to reason at length about the relationships between a large number of factors without losing detail. The used LLMs seem to be capable of identifying only key relationships between factors.

7 Conclusion

In this work, we proposed a novel method of LLM usage for BN structure learning, which initializes several LLMs with different experiences, independently queries them about the structure of the BN, and obtains the final structure of the BN by majority voting. We study the performance of this and another method on widely and not widely known BNs of different sizes. Moreover, we proposed a simple approach to check the contamination of BN in LLM, which shows that some widely known BNs are unsuitable for testing the LLM usage for BN structure elicitation. Our experiments showed that certain widely known BNs are also unsuitable for usage with such structure learning methods because their nodes are impossible to understand or their size is too big for LLM context size. The experiments on the other BNs show that our method performs better than the existing method with one of the three studied LLMs; however, both of the methods suffer a significant drop in performance when the size of BNs increases.

Limitations

The main limitation of this work is related to the engagement of LLMs to the experiments. First, we only used two proprietary and one open-sourced LLMs that were available by the date of the experiments. Currently, more LLMs are available (e.g. Llama 3, gpt-4o) that can handle the problem of insufficient context size. Second, no methods for LLMs fine-tuning were used. We leave these tasks for future work.

Acknowledgements

This paper is part of the R+D+i project TED2021-130295B-C33, funded by MCIN/AEI/10.13039/501100011033/ and by the "European Union NextGenerationEU/PRTR". The authors also acknowledge the support of the Galician Ministry for Education, Universities and Professional Training and the "ERDF A way of making Europe" through grants "Centro de investigación de Galicia accreditation 2024-2027 ED431G-2023/04" and "Reference Competitive Group accreditation 2022-2025 ED431C 2022/19".

References

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

Cedric Baudrit, Patrice Buche, Nadine Leconte, Christophe Fernandez, Maëllis Belna, and Geneviève Gésan-Guiziou. 2022. Decision support tool for the agri-food sector using data annotated by ontology and bayesian network: A proof of concept applied to milk microfiltration. *International Journal of Agricultural and Environmental Information Systems (IJAEIS)*, 13(1):1–22.

Shahab Behjati and Hamid Beigy. 2020. Improved K2 algorithm for bayesian network structure learning. *Engineering Applications of Artificial Intelligence*, 91:103617.

Aparna Elangovan, Jiayuan He, and Karin Verspoor. 2021. Memorization vs. generalization: quantifying data leakage in nlp performance evaluation. *arXiv preprint arXiv:2102.01818*.

Chen Gao, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li. 2023. Social-network simulation system with large language model-empowered agents. *arXiv preprint arXiv:2307.14984*.

Shahriar Golchin and Mihai Surdeanu. 2023a. Data contamination quiz: A tool to detect and estimate contamination in large language models. *arXiv preprint arXiv:2311.06233*.

Shahriar Golchin and Mihai Surdeanu. 2023b. Time travel in LLMs: Tracing data contamination in large language models. *arXiv preprint arXiv:2308.08493*.

Hideki Hamayasu, Masashi Miyao, Chihiro Kawai, Toshio Osamura, Akira Yamamoto, Hirozo Minami, Hitoshi Abiru, Keiji Tamaki, and Hirokazu Kotani. 2022. A proof-of-concept study to construct bayesian network decision models for supporting the categorization of sudden unexpected infant death. *Scientific Reports*, 12(1):9773.

Alon Jacovi, Avi Caciularu, Omer Goldman, and Yoav Goldberg. 2023. Stop uploading test data in plain text: Practical strategies for mitigating data contamination by evaluation benchmarks. *arXiv preprint arXiv:2305.10160*.

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, Matthias Stadler, Jochen Weller, Jochen Kuhn, and Gjergji Kasneci. 2023. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103:102274.

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.

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

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023a. Camel: Communicative agents for" mind" exploration of large scale language model society. *arXiv preprint arXiv:2303.17760*.

Ruosen Li, Teerth Patel, and Xinya Du. 2023b. Prd: Peer rank and discussion improve large language model based evaluations. *arXiv preprint arXiv:2307.02762*.

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, Alexandre Piché, ServiceNow Research, et al. 2023b. Can large language models build causal graphs? *arXiv preprint arXiv:2303.05279*.

R Thomas McCoy, Shunyu Yao, Dan Friedman, Matthew Hardy, and Thomas L Griffiths. 2023. Embers of autoregression: Understanding large language models through the problem they are trained to solve. *arXiv preprint arXiv:2309.13638*.

AE Nicholson, Steven Mascaro, Shreshth Thakur, KB Korb, and Ross Ashman. 2016. Delphi elicitation for strategic risk assessment. *preparation; available for review upon request*.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of GPT-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*.

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. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Yonatan Oren, Nicole Meister, Niladri Chatterji, Faisal Ladhak, and Tatsunori B Hashimoto. 2023. Proving test set contamination in black box language models. *arXiv preprint arXiv:2310.17623*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Jessica A Ramsay, Steven Mascaro, Anita J Campbell, David A Foley, Ariel O Mace, Paul Ingram, Meredith L Borland, Christopher C Blyth, Nicholas G Larkins, Tim Robertson, et al. 2022. Urinary tract infections in children: building a causal model-based decision support tool for diagnosis with domain knowledge and prospective data. *BMC Medical Research Methodology*, 22(1):218.

Gene Rowe, George Wright, and Fergus Bolger. 1991. Delphi: a reevaluation of research and theory. *Technological forecasting and social change*, 39(3):235–251.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023a. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10776–10787, Singapore. Association for Computational Linguistics.

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

Lydie Samie, Christophe Champod, Séverine Delémont, Patrick Basset, Tacha Hicks, and Vincent Castella. 2022. Use of bayesian networks for the investigation of the nature of biological material in casework. *Forensic Science International*, 331:111174.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2023. Detecting pretraining data from large language models. *arXiv preprint arXiv:2310.16789*.

Gabriele Sottocornola, Sanja Baric, Fabio Stella, and Markus Zanker. 2023. Development of a knowledge-based expert system for diagnosing post-harvest diseases of apple. *Agriculture*, 13(1):177.

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*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

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*.

Murray Turoff and Harold A Linstone. 2002. The delphi method-techniques and applications.

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*.

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

Zhenhailong Wang, Shaoguang Mao, Wenshan Wu, Tao Ge, Furu Wei, and Heng Ji. 2023b. Unleashing the emergent cognitive synergy in large language models: A task-solving agent through multi-persona self-collaboration. *arXiv preprint arXiv:2307.05300*.

Zidong Wang, Xiaoguang Gao, Yu Yang, Xiangyuan Tan, and Daqing Chen. 2021. Learning bayesian networks based on order graph with ancestral constraints. *Knowledge-Based Systems*, 211:106515.

Kai Xiong, Xiao Ding, Yixin Cao, Ting Liu, and Bing Qin. 2023. Examining the inter-consistency of large language models: An in-depth analysis via debate. Association for Computational Linguistics.

Mingchen Zhuge, Haozhe Liu, Francesco Faccio, Dylan R Ashley, Róbert Csordás, Anand Gopalakrishnan, Abdullah Hamdi, Hasan Abed Al Kader Hammoud, Vincent Herrmann, Kazuki Irie, et al. 2023. Mindstorms in natural language-based societies of mind. *arXiv preprint arXiv:2305.17066*.

Enrico Zio, Maryam Mustafayeva, and Andrea Montanaro. 2022. A bayesian belief network model for the risk assessment and management of premature screen-out during hydraulic fracturing. *Reliability Engineering & System Safety*, 218:108094.

A Our method overview

A.1 Facilitator prompts

System message

We are going to collect a Bayesian Network using a special communication protocol. The protocol is based on the paper "BARD: A Structured Technique for Group Elicitation of Bayesian Networks to Support Analytic Reasoning". It assumes that several specialists possess the necessary skills in the Bayesian Networks problem domain, and respond to our questions independently. Then we match their responses and help them to discuss the answers in an anonymous mode if any disagreements are found until a collective agreement is achieved.

First prompt requesting to think about possible profiles of the experts

We are going to collect a Bayesian network that requires some knowledge about *knowledge area* Here is the general idea of the Bayesian Network: *main task of the BN*. We will use another Large Language Model as experts. We will need 9 profiles of the experts that will be used to initialize the system message of the Language Model. The profiles must be as diverse as possible but at the same time, they must jointly possess all necessary knowledge to fulfill the task of knowledge elicitation for Bayesian Network collection. Think step-by-step what are the main qualities such experts should possess.

Second prompt requesting to generate a valid JSON with the profiles of the experts

Now please propose to me 9 profiles of the experts that will be used to initialize the system message of the Language Model. Turn your answer into JSON of the following form. Obligatory use such json from and do not include any side comments ‘‘‘json {"expert_1": "textual description of expert" (simply copy paste the details you used in the previous reply), "expert_2": "textual description of expert"(simply copy paste the details you used in the previous reply), "expert_3": "textual description of expert" (simply copy paste the details you used in the previous reply) ... } ‘‘‘

A.2 LLM expert prompts

System message

You will generate a predictive model using a specialized communication protocol. Assume the presence of multiple specialists possessing the required skills in the designated problem domain. Each specialist responds independently to our questions. Provide input as an expert with the following profile: *profile of the expert*.

First prompt demonstrating the list of explicit names of the BN nodes and requesting to reason about possible causal relations between them.

Consider the factors associated with the predictive model, represented by the list of nodes:*list of explicit names of BN nodes*. Now, analyze the relationships between these factors.

There are three possible types of relations:

- Factor A directly affects Factor B
- Factor B directly affects Factor A
- No direct effect between the two factors

Please systematically evaluate the interconnections between the specified factors, focusing only on significant relations

Second prompt requesting to summarize the generated causal relationships into a valid JSON

Summarize your thoughts in valid JSON format based on the relationships between the specified factors: *list of explicit names of BN nodes*. Use the following format to indicate connections between factors A and B: [(factor A, factor B)] (indicating that A directly affects B). Obligatorily keep the original names of the specified factors, do not change any letter from them. Provide only the valid JSON representation without additional discussion, following this structure:

[[factor A, factor B], (meaning the factor A directly affects factor B)

[factor C, factor E], (meaning the factor C directly affects factor E)

[factor D, factor H], (meaning the factor D directly affects factor H)

..... [factor .., factor ...]]

Decycling prompt

You have indicated that the changes of factor *node A* directly causally affect the changes of factor *node B* and the changes of factor *node B* directly causally affect the changes of factor *node A* in the same time. Please note that this is not possible in terms of the model that we develop. Consider possible ways to address this issue, determining which factor has a stronger effect on the other. Systematically think through the resolution and at the end summarize your thoughts in valid JSON of one of the following forms:

If you believe that the changes of factor *node A* directly causally affect the changes of factor *node B* more than the changes of factor *node B* directly causally affect the changes of factor *node A* output the JSON of form

"'json [*node A* , *node B*] "'

OR

If you believe that the changes of factor *node B* directly causally affect the changes of factor *node A* more than the changes of factor *node A* directly causally affect the changes of factor *node B* output the JSON of form

"'json [*node B* , *node A*] "'

B Harness method overview

Prompt Understand

You are an expert on Domain. You are investigating the cause and effect relationships between the following variables in your filed. Variable abbreviations and values are presented as follows.

Please understand the real meaning of each variable according to their possible values, and explain them in order.

variable sym1, values val1 · · · variable symn, values valn

Prompt Causal Discovery:

Based on the meaning of variables you provide, analyze the cause-and-effect relationships between them. Please give the results as a directed graph network in the form of <edge>variable1→variable2< edge>. Make sure that each edge represent a direct causality between the two variables.

Prompt Revsion:

Based on your explanation, check whether the following causal statements are correct, and give the reasons.

$xi1 \rightarrow xj1 \cdots xim \rightarrow xjm$

C Data contamination

C.1 Prompts

Node names request

Generate a list of nodes in the Bayesian Network discussed in the paper *paper name* and also available on *URL*. Provide details on each node and its role within the network structure.

BN structure request

Now retrieve the edges connecting the previously mentioned nodes in the Bayesian Network described in paper *paper name* and also available on *URL*. Express the network structure using the 'A->B' notation, indicating the presence of an edge from node A to node B in the Bayesian Network.

C.2 Example of answers

C.2.1 Answers about alarm BN

GPT-4 and alarm BN: *the data is very likely to be contaminated, because the answer covers almost all the nodes from bnlearn website and the names of the nodes are almost always similar to the ones on bnlearn.*

User ptompt

Generate a list of nodes in the Bayesian Network discussed in paper "I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine, pages 247-256. Springer-Verlag, 1989." and also available on bnlearn.com/bnrepository/discrete-medium.html. Provide details on each node and its role within the network structure.

System reply

The ALARM (A Logical Alarm Reduction Mechanism) network is a Bayesian network that represents a model of patient monitoring in intensive care units. It consists of 37 nodes and 46 edges. Here is a list of the nodes and their roles within the network:

1. History: Represents the patient's history of lung diseases.

2. CVP: Central Venous Pressure, a measure of blood pressure in the heart.

3. PCWP: Pulmonary Capillary Wedge Pressure, a measure of pressure in the heart.

4. Hypovolemia: A condition of decreased blood volume.

5. LVEDVolume: Left Ventricular End-Diastolic Volume, a measure of the amount of blood in the

heart.

6. StrokeVolume: The amount of blood pumped by the left ventricle of the heart in one contraction.

7. CO: Cardiac Output, the amount of blood the heart pumps through the circulatory system in a minute.

8. BP: Blood Pressure.

9. TPR: Total Peripheral Resistance, the resistance blood must overcome to be able to move within the body.

10. Anaphylaxis: A severe, potentially life-threatening allergic reaction.

11. InsuffAnesth: Insufficient Anesthesia.

12. TPR: Total Peripheral Resistance.

13. Anaphylaxis: Severe allergic reaction.

14. MinVol: Minimum Volume.

15. FIO2: Fraction of Inspired Oxygen, the concentration of oxygen in the inhaled air.

16. PVSat: Peripheral Venous Saturation, a measure of the oxygen saturation in the peripheral veins.

17. SaO2: Arterial Oxygen Saturation, a measure of the amount of oxygen-carrying hemoglobin in the blood.

18. PulmShunt: Pulmonary Shunt, a condition that allows blood to bypass the lungs.

19. Intubation: The process of inserting a tube through the mouth and then into the airway.

20. Press: Pressure.

21. Disconnect: Disconnection of the patient from the ventilator.

22. MinVolSet: Minimum Volume Set.

23. VentMach: Ventilation Machine.

24. KinkedTube: Kinked Tube.

25. VentTube: Ventilation Tube.

26. VentLung: Ventilation Lung.

27. VentAlv: Ventilation Alveoli.

28. ArtCO2: Arterial CO2 Pressure.

29. CO2: Carbon Dioxide.

30. HR: Heart Rate.

31. Catechol: Catecholamines, hormones made by the adrenal glands.

32. HRBP: Heart Rate Blood Pressure.

33. ErrLowOutput: Error Low Output.

34. HREKG: Heart Rate EKG.

35. HRSat: Heart Rate Oxygen Saturation.

36. InsuffAnesth: Insufficient Anesthesia.

37. ExpCO2: Expired CO2.

Each node represents a medical condition, a medical measurement, or a medical procedure. The edges between the nodes represent causal relationships between these conditions, measurements, and procedures. For example, the node "Hypovolemia" (decreased blood volume) has edges to nodes "CVP" (Central Venous Pressure) and "PCWP" (Pulmonary Capillary Wedge Pressure), indicating that hypovolemia can cause changes in these measurements.

User ptompt

Now retrieve the edges connecting the previously mentioned nodes in the Bayesian Network described in paper "I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine, pages 247-256. Springer-Verlag, 1989." and also available on bnlearn.com/bnrepository/discrete-medium.html. Express the network structure using the 'A->B' notation, indicating the presence of an edge from node A to node B in the Bayesian Network.

System reply

The ALARM network is a complex Bayesian network with 46 edges. Here are the edges connecting the nodes in the network, expressed in the 'A->B' notation:

1. History->LVEDVolume
2. History->CVP
3. History->PCWP
4. CVP->LVEDVolume
5. PCWP->LVEDVolume
6. Hypovolemia->CVP
7. Hypovolemia->PCWP
8. LVEDVolume->StrokeVolume
9. StrokeVolume->CO
10. Hypovolemia->CO
11. CO->BP
12. TPR->BP
13. Anaphylaxis->TPR
14. Anaphylaxis->InsuffAnesth
15. InsuffAnesth->TPR
16. TPR->BP
17. Anaphylaxis->MinVol
18. MinVol->FIO2
19. FIO2->PVSat
20. PVSat->SaO2
21. SaO2->PulmShunt
22. PulmShunt->Intubation
23. Intubation->Press
24. Press->Disconnect
25. Disconnect->MinVolSet

26. MinVolSet->VentMach
27. VentMach->KinkedTube
28. KinkedTube->VentTube
29. VentTube->VentLung
30. VentLung->VentAlv
31. VentAlv->ArtCO2
32. ArtCO2->CO2
33. CO2->HR
34. HR->Catechol
35. Catechol->HRBP
36. HRBP->ErrLowOutput
37. ErrLowOutput->HREKG
38. HREKG->HRSat
39. HRSat->InsuffAnesth
40. InsuffAnesth->ExpCO2
41. ExpCO2->KinkedTube
42. KinkedTube->MinVolSet
43. MinVolSet->VentMach
44. VentMach->VentTube
45. VentTube->VentLung
46. VentLung->VentAlv

Each edge represents a causal relationship between two nodes in the network. For example, the edge 'Hypovolemia->CVP' indicates that hypovolemia (decreased blood volume) can cause changes in CVP (Central Venous Pressure).

GPT-3.5 and alarm BN: *the data is also likely to be contaminated because the names of the nodes are similar to the ones on bnlearn website*

User prompt

Generate a list of nodes in the Bayesian Network discussed in paper "I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine, pages 247-256. Springer-Verlag, 1989." and also available on bnlearn.com/bnrepository/discrete-medium.html. Provide details on each node and its role within the network structure.

System reply

The Bayesian Network discussed in the paper "The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks" consists of the following nodes:

1. HISTORY - Role: Represents the patient's history of similar conditions. - Description: This node captures the patient's medical history and is used to assess the likelihood of the current condition based on past occurrences.

2. CVP - Role: Represents the central venous pressure. - Description: This node captures the patient's central venous pressure, which is an important indicator of the patient's cardiovascular status.

3. PCWP - Role: Represents the pulmonary capillary wedge pressure. - Description: This node captures the patient's pulmonary capillary wedge pressure, which is an important indicator of the patient's cardiac function.

4. HYPOTENSION - Role: Represents the presence of hypotension. - Description: This node captures the presence of hypotension in the patient, which is a critical condition that requires immediate attention.

5. LVEDVOLUME - Role: Represents the left ventricular end-diastolic volume. - Description: This node captures the patient's left ventricular end-diastolic volume, which is an important indicator of the patient's cardiac function.

6. LVFAILURE - Role: Represents the presence of left ventricular failure. - Description: This node captures the presence of left ventricular failure in the patient, which is a critical condition that requires immediate attention.

7. STROKEVOLUME - Role: Represents the stroke volume. - Description: This node captures the patient's stroke volume, which is an important indicator of the patient's cardiac function.

8. ERRLOWOUTPUT - Role: Represents the presence of low cardiac output. - Description: This node captures the presence of low cardiac output in the patient, which is a critical condition that requires immediate attention.

9. HRBP - Role: Represents the arterial blood pressure. - Description: This node captures the patient's arterial blood pressure, which is an important indicator of the patient's cardiovascular status.

10. HREKG - Role: Represents the heart rate from an EKG. - Description: This node captures the patient's heart rate as measured by an electrocardiogram, which is an important indicator of the patient's cardiac function.

These nodes collectively form the Bayesian Network used in the ALARM Monitoring System to assess the patient's cardiovascular and cardiac status and make inferences about potential critical conditions.

User prompt

Now retrieve the edges connecting the previously mentioned nodes in the Bayesian Network described in paper "I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The

ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine, pages 247-256. Springer-Verlag, 1989." and also available on bnlearn.com/bnrepository/discrete-medium.html. Express the network structure using the 'A->B' notation, indicating the presence of an edge from node A to node B in the Bayesian Network.

System reply

The edges connecting the nodes in the Bayesian Network described in the paper "The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks" are as follows:

1. HISTORY -> HYPOTENSION
2. CVP -> LVFAILURE
3. PCWP -> LVFAILURE
4. HYPOTENSION -> LVFAILURE
5. LVEDVOLUME -> LVFAILURE
6. LVFAILURE -> ERRLOWOUTPUT
7. LVFAILURE -> STROKEVOLUME
8. LVFAILURE -> HREKG
9. LVFAILURE -> HREKG
10. LVFAILURE -> HRBP

These edges represent the relationships and dependencies between the nodes in the Bayesian Network, indicating the flow of probabilistic influence and information within the network structure.

Llama2 and alarm BN: *the data is unlikely to be contaminated because the proposed nodes do not coincide with almost any of the nodes from the BN on bnlearn website*

User prompt

Generate a list of nodes in the Bayesian Network discussed in paper "I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine, pages 247-256. Springer-Verlag, 1989." and also available on bnlearn.com/bnrepository/discrete-medium.html. Provide details on each node and its role within the network structure.

System reply

The Bayesian Network (BN) discussed in the paper "The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks" by I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper (1989) is a discrete Bayesian Network that models the probability distribution over 15 variables (nodes) related to the monitoring of a chemical process. The network is available on bnlearn.com/bnrepository/discrete-medium.html.

Here's a list of the nodes in the network, along with a brief description of each node and its role within the network structure:

1. Alarm: This node represents the alarm status of the chemical process. It has two possible values: "alarm" (A) and "no alarm" (not A). The probability distribution over this node is updated based on the values of the other nodes in the network.

2. Pressure: This node represents the pressure reading of a pressure sensor in the chemical process. It has three possible values: "low" (P_L), "normal" (P_N), and "high" (P_H). The probability distribution over this node is updated based on the values of the other nodes in the network.

3. Flow: This node represents the flow rate of the chemical process. It has three possible values: "low" (F_L), "normal" (F_N), and "high" (F_H). The probability distribution over this node is updated based on the values of the other nodes in the network.

4. Level: This node represents the level of a tank in the chemical process. It has three possible values: "low" (L), "normal" (N), and "high" (H). The probability distribution over this node is updated based on the values of the other nodes in the network.

5. Temperature: This node represents the temperature reading of a temperature sensor in the chemical process. It has three possible values: "low" (T_L), "normal" (T_N), and "high" (T_H). The probability distribution over this node is updated based on the values of the other nodes in the network.

6. Conductivity: This node represents the conductivity reading of a conductivity sensor in the chemical process. It has three possible values: "low" (C_L), "normal" (C_N), and "high" (C_H). The probability distribution over this node is updated based on the values of the other nodes in the network.

7. pH: This node represents the pH reading of a pH sensor in the chemical process. It has three possible values: "acidic" (pH_A), "normal" (pH_N), and "basic" (pH_B). The probability distribution over this node is updated based on the values of the other nodes in the network.

8. Oxygen: This node represents the oxygen

reading of an oxygen sensor in the chemical process. It has three possible values: "low" (O_L), "normal" (O_N), and "high" (O_H). The probability distribution over this node is updated based on the values of the other nodes in the network.

9. Nitrogen: This node represents the nitrogen reading of a nitrogen sensor in the chemical process. It has three possible values: "low" (N2_L), "normal" (N2_N), and "high" (N2_H). The probability distribution over this node is updated based on the values of the other nodes in the network.

10. Water: This node represents the water reading of a water sensor in the chemical process. It has three possible values: "low" (W_L), "normal" (W_N), and "high" (W_H). The probability distribution over this node is updated based on the values of the other nodes in the network.

11. Air: This node represents the air reading of an air sensor in the chemical process. It has three possible values: "low" (A_L), "normal" (A_N), and "high" (A_H). The probability distribution over this node is updated based on the values of the other nodes in the network.

12. Steam: This node represents the steam reading of a steam sensor in the chemical process. It has three possible values: "low" (S_L), "normal" (S_N), and "high" (S_H). The probability distribution over this node is updated based on the values of the other nodes in the network.

13. Valve1: This node represents the status of a valve in the chemical process. It has two possible values: "open" (V1_O) and "closed" (V1_C). The probability distribution over this node is updated based on the values of the other nodes in the network.

14. Valve2: This node represents the status of a second valve in the chemical process. It has two possible values: "open" (V2_O) and "closed" (V2_C). The probability distribution over this node is updated based on the values of the other nodes in the network.

15. Pump: This node represents the status of a pump in the chemical process. It has two possible values: "on" (P_O) and "off" (P_C). The probability distribution over this node is updated based on the values of the other nodes in the network.

The Bayesian Network is structured such that each node is connected to other nodes in the network, representing the conditional dependencies between the variables. For example, the probability distribution over the "Alarm" node is updated based on the values of the "Pressure," "Flow," "Level," "Temperature," "Conductivity," "pH," "Oxygen," "Nitrogen," "Water," "Air," and "Steam" nodes. Similarly, the probability distribution over the "Pressure" node is updated based on the values of the "Flow," "Level," "Temperature," and "Conductivity" nodes.

The network uses a directed acyclic graph (DAG) structure to represent the conditional dependencies between the variables. The direction of the arrows in the graph indicates the direction of causality between the variables. The probability distributions over each node are updated using Bayes' rule, which combines the prior probability distribution over the node with the likelihood of the node given the values of its parent nodes in the DAG.

Overall, the Bayesian Network provides a probabilistic model of the chemical process that can be used to reason about the likelihood of different states of the process given the available sensor readings. The network can also be used to identify potential faults or anomalies in the process and to make predictions about future states of the process based on historical data.

User prompt

Now retrieve the edges connecting the previously mentioned nodes in the Bayesian Network described in paper "I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine, pages 247-256. Springer-Verlag, 1989." and also available on bnlearn.com/bnrepository/discrete-medium.html. Express the network structure using the 'A->B' notation, indicating the presence of an edge from node A to node B in the Bayesian Network.

System reply

Sure! Here are the edges connecting the nodes in the Bayesian Network described in the paper "The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks" by I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper:

* Alarm -> Pressure
* Alarm -> Flow
* Alarm -> Level
* Alarm -> Temperature
* Alarm -> Conductivity
* Alarm -> pH
* Alarm -> Oxygen

* Alarm -> Nitrogen
* Alarm -> Water
* Alarm -> Air
* Alarm -> Steam
* Pressure -> Flow
* Pressure -> Level
* Pressure -> Temperature
* Pressure -> Conductivity
* Flow -> Level
* Flow -> Temperature
* Flow -> Conductivity
* Level -> Temperature
* Level -> Conductivity
* Temperature -> Conductivity
* Conductivity -> pH
* Conductivity -> Oxygen
* Conductivity -> Nitrogen
* Conductivity -> Water
* Conductivity -> Air
* Conductivity -> Steam

Here's the network structure using the 'A->B' notation:

* Alarm -> Pressure
* Alarm -> Flow
* Alarm -> Level
* Alarm -> Temperature
* Alarm -> Conductivity
* Alarm -> pH
* Alarm -> Oxygen
* Alarm -> Nitrogen
* Alarm -> Water
* Alarm -> Air
* Alarm -> Steam
* Pressure -> Flow
* Pressure -> Level
* Pressure -> Temperature
* Pressure -> Conductivity
* Flow -> Level
* Flow -> Temperature
* Flow -> Conductivity
* Level -> Temperature
* Level -> Conductivity
* Temperature -> Conductivity
* Conductivity -> pH
* Conductivity -> Oxygen
* Conductivity -> Nitrogen
* Conductivity -> Water
* Conductivity -> Air
* Conductivity -> Steam

Note that the network structure is a directed acyclic graph (DAG), which means that there are no cycles in the graph and the edges have a direction, indicating the direction of causality between the variables.

C.2.2 Answers about non-existing BN

GPT-4 and non-existing BN: *Can not find the paper and warns that the list of nodes will be the hypothetical.*

User prompt

Generate a list of nodes in the Bayesian Network discussed in paper "Frozen Paths: Leveraging Bayesian Networks for Enhanced Risk Analysis in Severe Freeze Weather during Road Construction". Provide details on each node and its role within the network structure.

System reply

The paper "Frozen Paths: Leveraging Bayesian Networks for Enhanced Risk Analysis in Severe Freeze Weather during Road Construction" does not seem to exist, as multiple searches in academic databases yield no results. Therefore, it's impossible to provide a list of nodes in the Bayesian Network discussed in this paper.

However, in a hypothetical Bayesian Network for risk analysis in severe freeze weather during road construction, the nodes could represent various factors affecting the construction process. Here are some potential nodes:

1. Weather Conditions: This node could represent the probability of severe freeze weather. It would be a parent node to many other nodes, as weather conditions can affect many aspects of road construction.

2. Construction Materials: This node could represent the type and quality of materials used in construction. The performance of these materials could be affected by severe freeze weather.

3. Worker Safety: This node could represent the risk of accidents or injuries to workers due to severe freeze weather.

4. Construction Equipment: This node could represent the reliability and performance of construction equipment in severe freeze weather.

5. Project Delays: This node could represent the probability of the construction project being delayed due to severe freeze weather.

6. Cost Overruns: This node could represent the probability of the construction project going over budget due to severe freeze weather.

7. Quality of Construction: This node could represent the overall quality of the construction

project, which could be affected by severe freeze weather.

Each of these nodes would have various parent and child nodes, creating a complex network of dependencies and probabilities. The Bayesian Network would be used to analyze and predict the overall risk of the construction project in the face of severe freeze weather.

GPT-3.5 and non-existing BN: *Generates the list of nodes (without any warnings)*

User prompt

Generate a list of nodes in the Bayesian Network discussed in paper "Frozen Paths: Leveraging Bayesian Networks for Enhanced Risk Analysis in Severe Freeze Weather during Road Construction". Provide details on each node and its role within the network structure.

System reply

1. Weather Conditions - This node represents the different weather conditions that can occur during road construction, such as severe freeze, snow, ice, etc. It plays a crucial role in determining the risk of severe freeze weather affecting the construction site.

2. Road Construction Activities - This node represents the various activities involved in road construction, such as laying asphalt, pouring concrete, etc. It influences the likelihood of severe freeze weather impacting the construction process.

3. Risk of Severe Freeze Weather - This node represents the likelihood of severe freeze weather occurring at the construction site. It is influenced by the weather conditions and other environmental factors.

4. Impact on Construction Schedule - This node represents the potential impact of severe freeze weather on the construction schedule. It is influenced by the risk of severe freeze weather and road construction activities.

5. Safety Risks - This node represents the safety risks associated with severe freeze weather during road construction. It is influenced by the impact on construction schedule and other factors.

6. Mitigation Strategies - This node represents the various strategies that can be implemented to mitigate the impact of severe freeze weather on road construction. It is influenced by the risk of severe freeze weather and safety risks.

7. Cost Implications - This node represents the potential cost implications of severe freeze weather on road construction. It is influenced by the impact on construction schedule and mitigation strategies.

These nodes collectively form a Bayesian Network that allows for enhanced risk analysis of severe freeze weather during road construction, taking into account various factors and their interdependencies.

Llama2 and non-existing BN: *Generates the list of nodes (without any warnings)*

User prompt

Generate a list of nodes in the Bayesian Network discussed in paper "Frozen Paths: Leveraging Bayesian Networks for Enhanced Risk Analysis in Severe Freeze Weather during Road Construction". Provide details on each node and its role within the network structure.

System reply

The Bayesian Network discussed in the paper "Frozen Paths: Leveraging Bayesian Networks for Enhanced Risk Analysis in Severe Freeze Weather during Road Construction" is designed to assess the risk of frozen ground during road construction in cold regions. The network consists of 14 nodes, each representing a specific variable or factor that affects the risk of frozen ground. Here is a list of the nodes, along with a brief description of each node and its role within the network structure:

1. Air Temperature (AT): Represents the temperature of the air in degrees Celsius. AT is a primary input variable that affects the risk of frozen ground.

2. Ground Temperature (GT): Represents the temperature of the ground in degrees Celsius. GT is a primary output variable that indicates the likelihood of frozen ground.

3. Soil Moisture (SM): Represents the moisture content of the soil. SM affects the rate of heat transfer between the ground and the air, and therefore affects the risk of frozen ground.

4. Wind Speed (WS): Represents the wind speed in meters per second. WS affects the rate of heat transfer between the ground and the air, and therefore affects the risk of frozen ground.

5. Relative Humidity (RH): Represents the relative humidity of the air. RH affects the rate of heat transfer between the ground and the air, and therefore affects the risk of frozen ground.

6. Cloud Cover (CC): Represents the extent of cloud cover. CC affects the amount of solar radiation that reaches the ground, which in turn affects the risk of frozen ground.

7. Time of Day (TOD): Represents the time of day, which affects the amount of solar radiation that reaches the ground, and therefore affects the risk of frozen ground.

8. Day of Year (DOY): Represents the day of the year, which affects the amount of solar radiation that reaches the ground, and therefore affects the risk of frozen ground.

9. Shading (SH): Represents the extent of shading on the ground, which affects the amount of solar radiation that reaches the ground, and therefore affects the risk of frozen ground.

10. Slope (SL): Represents the slope of the ground, which affects the rate of heat transfer between the ground and the air, and therefore affects the risk of frozen ground.

11. Aspect (AS): Represents the aspect of the ground, which affects the amount of solar radiation that reaches the ground, and therefore affects the risk of frozen ground.

12. Vegetation (VG): Represents the extent of vegetation cover, which affects the moisture content of the soil and the rate of heat transfer between the ground and the air, and therefore affects the risk of frozen ground.

13. Soil Type (ST): Represents the type of soil, which affects the thermal properties of the ground and the rate of heat transfer between the ground and the air, and therefore affects the risk of frozen ground.

14. Weather Station (WSN): Represents the location of a weather station, which provides measurements of the weather variables (AT, WS, RH, CC, TOD, DOY, SH, SL, AS, VG, and ST) that affect the risk of frozen ground.

The nodes in the Bayesian Network are connected by directed arrows that represent conditional dependencies between the variables. The direction of the arrows indicates the direction of causality between the variables. For example, the arrow pointing from AT to GT indicates that air temperature affects ground temperature. The network structure allows for the propagation of uncertainty between the variables, enabling the estimation of the risk of frozen ground based on the input variables.

D Detailed statistics of the experiments

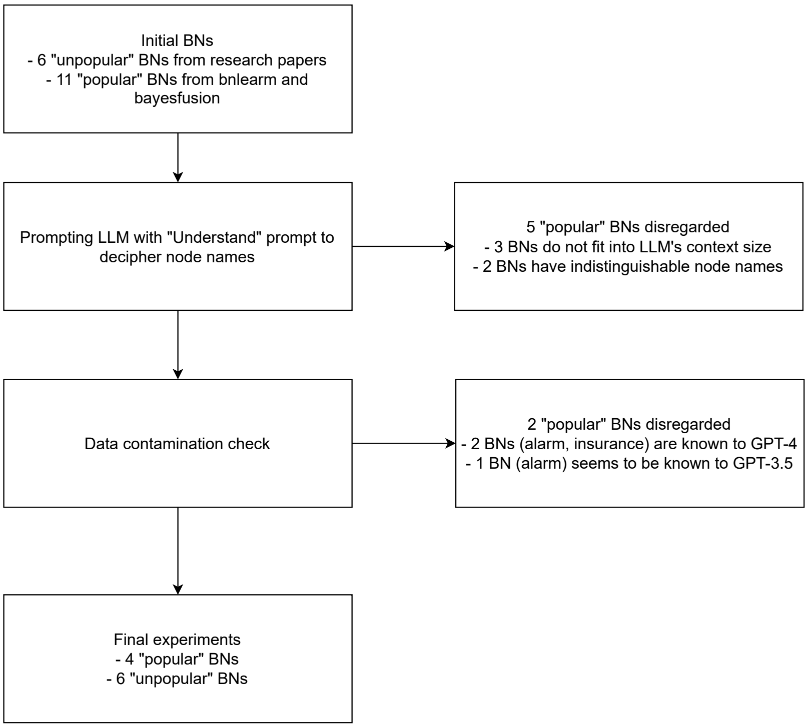

Figure 6: The process of BNs selection for final experiments.

BN	Method	GPT-3.5				Llama2			
		FP	FN	F-score	SHD/edg	FP	FN	F-score	SHD/edg
coma	har	0	2	0.85	0.4	4	4	0.5	1.6
	our	0	3	0.75	0.6	2	3	0.67	0.94
agro	har	3	8	0.54	1.1	1	7	0.65	0.8
	our	2	10	0.4	1.2	3	8	0.55	1.07
sperm	har	5	8	0.61	1.18	4	10	0.51	1.27
	our	3	8	0.61	1.06	1	8	0.65	0.87
screen	har	11	20	0.5	1.48	8	18	0.57	1.24
	our	3	20	0.52	1.07	10	18	0.56	1.33
covid	har	3	15	0.76	0.69	0	17	0.75	0.65
	our	1	21	0.64	0.86	4	26	0.48	1.14
sids	har	17	22	0.57	1.44	0	27	0.48	1
	our	3	23	0.58	0.97	5	23	0.59	1.03
apple	har	18	53	0.58	1.15	1	62	0.48	1.02
	our	13	40	0.71	0.86	24	60	0.5	1.36
hailfinder	har	70	66	0.49	1.24	5	64	0.52	1.05
	our	3	65	0.5	1.04	37	52	0.61	1.34
urinary	har	108	103	0.47	1.97	1	107	0.48	1.01
	our	2	99	0.55	0.94	3	105	0.5	1.01
hepar2	har	36	112	0.56	1.17	15	117	0.53	1.07
	our	13	117	0.54	1.05	23	116	0.53	1.13

Table 7: Detailed information about the performance of *harness* and our method with GPT-3.5 and Llama2. FP and FN correspond to false positive and false negative edges in the learned BN structure respectively.

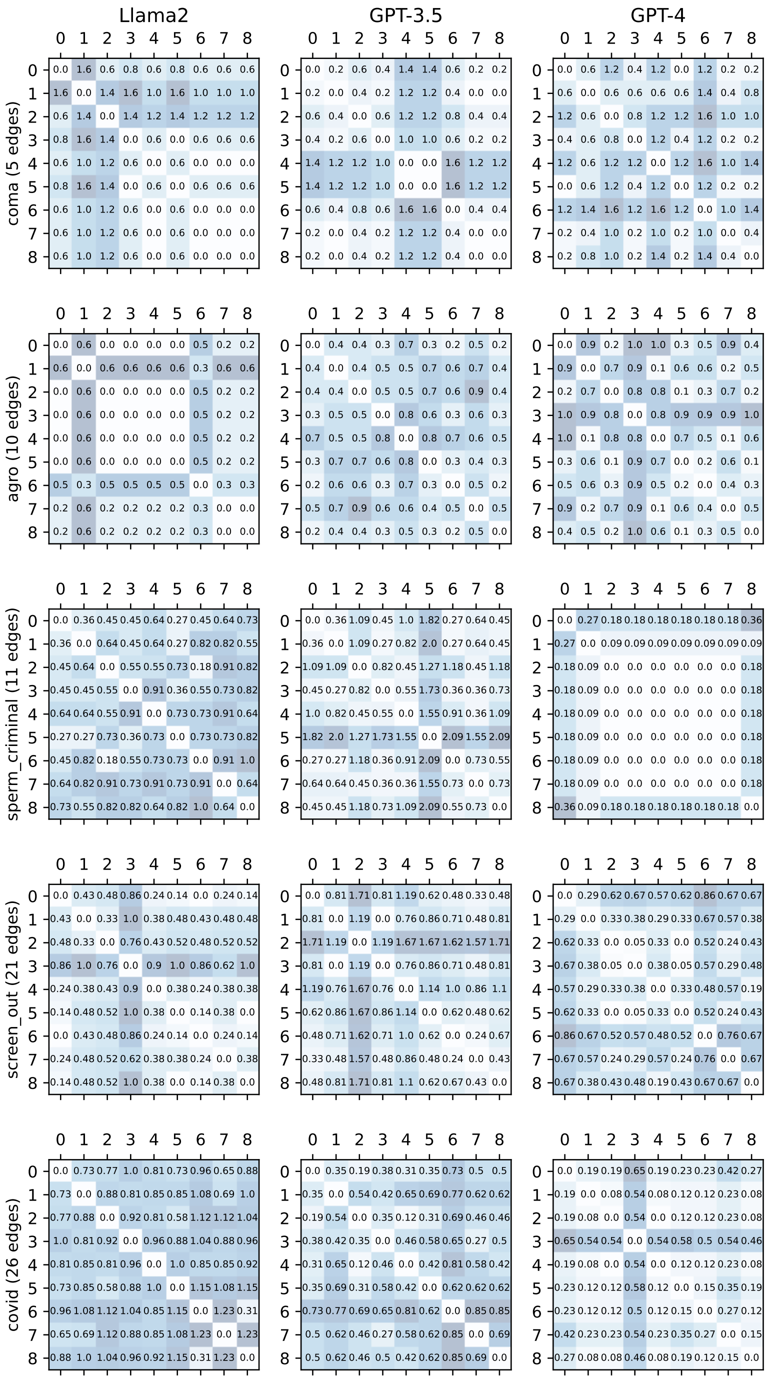

Figure 7: SHD between structures generated by ‘LLM-experts” of all types for all BNs engaged in the study. Plot 1

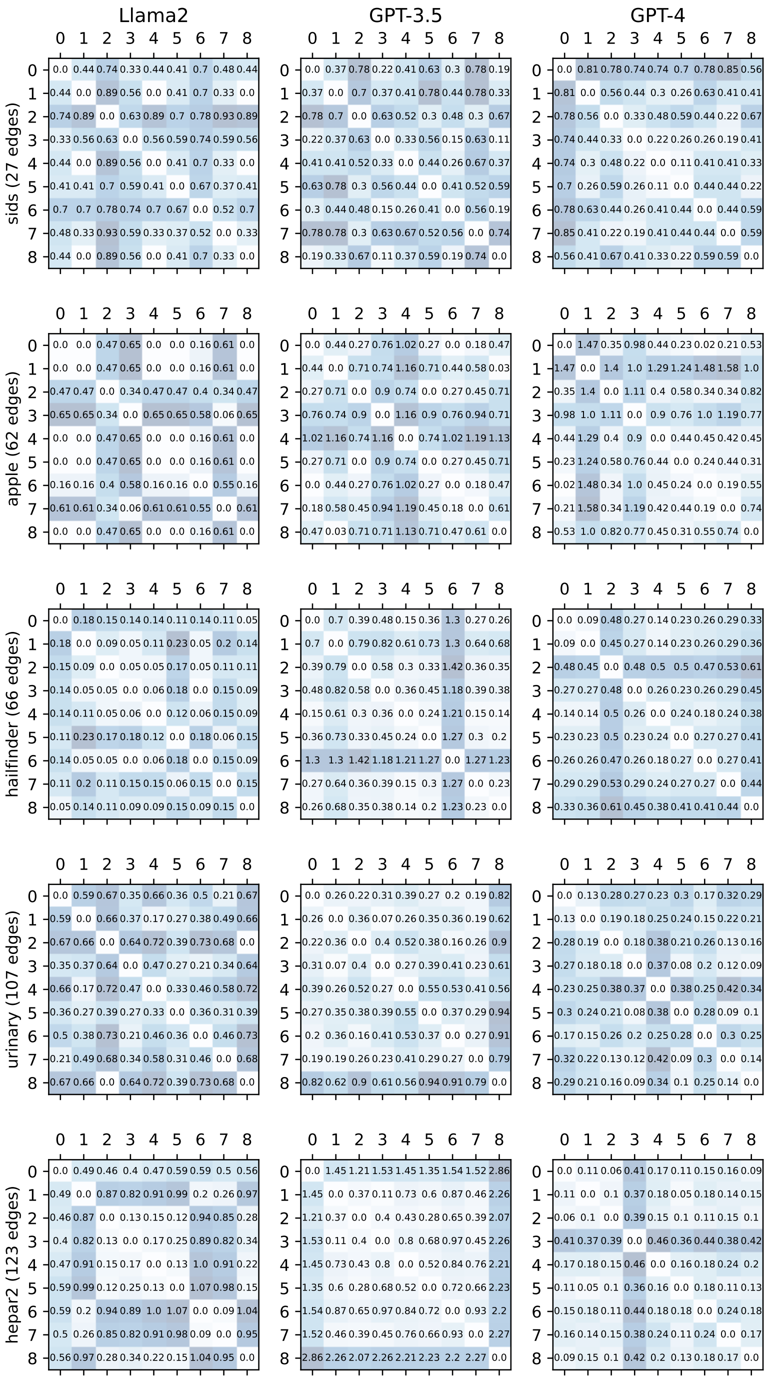

Figure 8: SHD between structures generated by ‘LLM-experts” of all types for all BNs engaged in the study. Plot 2.

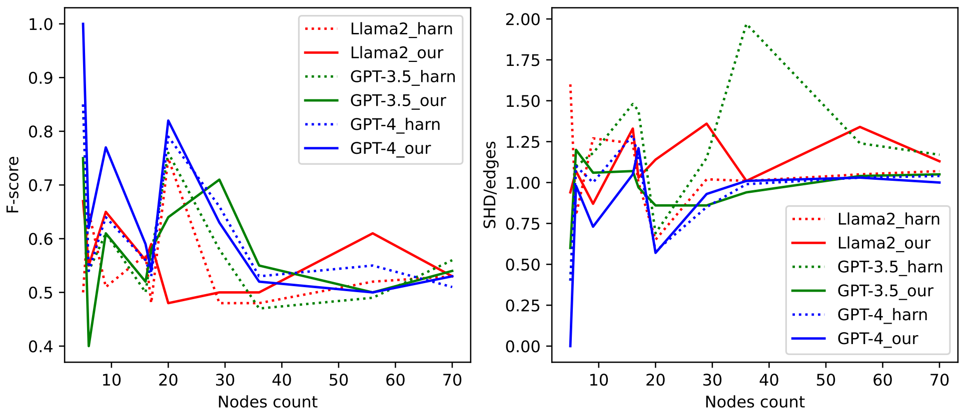

Figure 9: F-score and SHD resulting from both studied method with all LLMs related to the number of nodes in a BN.

E Error analysis

Llama2 generates BN structure using non-existing node obtained from its system message

System message

You will generate a predictive model using a specialized communication protocol. Assume the presence of multiple specialists possessing the required skills in the designated problem domain. Each specialist responds independently to our questions. Provide input as an expert with the following profile: Epidemiologist with 15 years of experience in studying the distribution and determinants of health-related events, diseases, or health-related characteristics among populations, including the epidemiology of **urinary tract infections** in children.

User prompt

Consider the factors associated with the predictive model, represented by the list of nodes < Age group >,< Sex >,< Comorbidity >,< Diarrhoea >,< On antibiotics upon presentation to ED >,< level of colonisation of the perineum/ external genitalia by E. coli >,< level of colonisation of the perineum/ external genitalia by non-E. coli gram negative bacteria >,< level of colonisation of the perineum/ external genitalia by gram positive bacteria >,< Pathogenic organism infecting the urinary tract and causing disease >,< Current clinical phenotype >,< Empiric antibiotic prescription received during the ED presentation >,< Urine collection method >,< Assessment of epithelial cells via microscopic or automated analysis >,< risk of a non-causative organism/s entering the urine specimen during the specimen collection process >,< Assessment of bacteria via microscopic analysis >,< isolation of E.coli from the urine specimen >,< isolation of non-E.coli gram negative bacteria from the urine specimen >,< isolation of gram positive bacteria from the urine specimen >,< leukocyte esterase detected on urine dipstick >,< Nitrite detected on the urine dipstick >,< assessment of leucocytes level in the urine specimen via microscopic analysis >,< Leukocyte counts in blood >,< C-Reactive protein in blood >,< Absolute neutrophil counts in blood >,< Genital pain (or discomfort), dysuria, and urinary tract relevant discomfortn >,< Abdominal pain >,< Haematuria >,< Foul smelling urine >,< Result of ultrasound investigation >,< Respiratory symptoms >,< Parent reported fever in medical notes >,< Temperature (degrees celsius) >,< Irritability >,< Lethargy >,< Nausea and or vomiting >,< Poor oral intake >. Now, analyze the relationships between these factors. There are three possible types of relations: Factor A directly affects Factor B Factor B directly affects Factor A No direct effect between two factors

System reply

....

User prompt

.....

System reply

Here is a valid JSON representation of the relationships between the specified factors:“‘json[["Age group", "**Urinary tract infection**"], ["Sex", "**Urinary tract infection**"], ["Comorbidity", "**Urinary tract infection**"], ["Diarrhea", "**Urinary tract infection**"], ["On antibiotics upon presentation to ED", "**Urinary tract infection**"], ["Level of colonisation of the perineum/external genitalia by E. coli", "**Urinary tract infection**"], ["Level of colonisation of the perineum/external genitalia by non-E. coli gram negative bacteria", "**Urinary tract infection**"], ["Level of colonisation of the perineum/external genitalia by gram positive bacteria", "**Urinary tract infection**"], ["Pathogenic organism infecting the urinary tract and causing disease", "**Urinary tract infection**"], ["Current clinical phenotype", "**Urinary tract infection**"], ["Empiric antibiotic prescription received during the ED presentation", "**Urinary tract infection**"], ["Urine

Method	Llama2	GPT-3.5	GPT-4
Harness	0	1.2	0
our	0.6	15.6	0.6

Table 8: Average number of cycles generated by both methods with all LLMs for all BNs

collection method", "**Urinary tract infection**"], ["Assessment of epithelial cells via microscopic or automated analysis", "**Urinary tract infection**"], ["Risk of a non-causative organism/s entering the urine specimen during the specimen collection process", "**Urinary tract infection**"], ["Assessment of bacteria via microscopic analysis", "**Urinary tract infection**"], ["Isolation of E.coli from the urine specimen", "**Urinary tract infection**"], ["Isolation of non-E.coli gram negative bacteria from the urine specimen", "**Urinary tract infection**"], ["Isolation of gram positive bacteria from the urine specimen", "**Urinary tract infection**"], ["Leukocyte esterase detected on urine dipstick", "**Urinary tract infection**"], ["Nitrite detected on the urine dipstick", "**Urinary tract infection**"], ["Assessment of leucocytes level in the urine specimen via microscopic analysis", "**Urinary tract infection**"], ["Leukocyte counts in blood", "**Urinary tract infection**"], ["C-Reactive protein in blood", "**Urinary tract infection**"], ["Absolute neutrophil counts in blood", "**Urinary tract infection**"], ["Genital pain (or discomfort), dysuria, and urinary tract relevant discomfort", "**Urinary tract infection**"], ["Abdominal pain", "**Urinary tract infection**"], ["Haematuria", "**Urinary tract infection**"], ["Foul smelling urine", "**Urinary tract infection**"], ["Result of ultrasound investigation", "**Urinary tract infection**"], ["Respiratory symptoms", "**Urinary tract infection**"], ["Parent reported fever in medical notes", "**Urinary tract infection**"], ["Temperature (degrees celsius)", "**Urinary tract infection**"], ["Irritability", "**Urinary tract infection**"], ["Lethargy", "**Urinary tract infection**"], ["Nausea and or vomiting", "**Urinary tract infection**"], ["Poor oral intake", "**Urinary tract infection**"]]"'Note that this representation only includes the direct relationships between the factors and does not include any indirect relationships or interactions between factors.

GPT-4 refuses to generate a JSON with urinary BN structure

System reply (to the second prompt of our method)

As a medical ethicist, I am not qualified to determine the direct relationships between these medical factors. My role is to ensure that the process of building this predictive model respects the ethical principles of autonomy, beneficence, non-maleficence, and justice. I recommend consulting with a medical professional or a data scientist to determine these relationships.

BN	Llama2	GPT-3.5	GPT-4
coma	3	6	0
agro	0	11	2
sperm	3	49	0
screen	0	60	1
covid	0	2	0
sids	0	23	0
apple	0	4	2
hailfinder	0	0	0
urinary	0	0	0
hepar2	0	1	1

Table 9: Cycles generated by LLMs for different BNs when using our method

Llama2 generates a "cause-and-effect chain" rather than BN structure (hailfinder BN)

System message

You will generate a predictive model using a specialized communication protocol. Assume the presence of multiple specialists possessing the required skills in the designated problem domain. Each specialist responds independently to our questions. Provide input as an expert with the following profile: Communication and collaboration expert with a background in psychology, sociology, or organizational behavior, skilled in facilitating effective communication and collaboration among diverse groups

User prompt

Consider the factors associated with the predictive model, represented by the list of nodes < relationship between the scenario and the morning convective inhibition >,< level of morning convective inhibition >,< morning convective inhibition in the given scenario >,< atmospheric moisture conditions >,< relationship between the scenario and the morning atmospheric instability >,< Lifted Index (LI) and density relationship >,< morning atmospheric instability in the given scenario >,< morning atmospheric instability >,< combined vertical motion >,< mesoscale area >,< combined moisture conditions >,< area of dry air >,< outflow from the mountains >,< wind hodograph >,< morning boundaries >,< overall boundaries >,< Composite Planetary Forecast >,< Convective Available Potential Energy (CAPE) Change >,< Convective Available Potential Energy (CAPE) in Scenario >,< Instability in Mountainous Terrain >,< Cloud Shade due to Convection >,< Combined Clouds >,< Other Cloud Shade >,< Visible Cloud Cover >,< Infrared Cloud Cover >,< Saturation Content Moisture >,< Rao Content Moisture >,< 0.7 Micrometer Vertical Motion >,< Subjective Vertical Motion >,< Quasi-Geostrophic Vertical Motion >,< Latest Convective Available Potential Energy (CAPE) Inhibition >,< Lifted Level Index Weighted >,< Current Propagation of Convection >,< Date of Weather Forecast >,< Weather Scenario >,< Atmospheric Dewpoints >,< Low-Level Moisture Advection >,< Instability Change >,< Instability Scale in Scenario >,< Low-Level Lapse Rate >,< Mean Relative Humidity >,< Mid-Level Lapse Rate >,< Mountainous Forecast >,< Movement Features >,< Scenario Relative to 3-4 >,< Plains Forecast >,< North 34-Star Forecast >,< Scenario Relative Planetary Forecast >,< Region 5 Forecast >,< Relative Humidity Ratio >,< Surface Wind Shift Distance >,< Synoptic Forcing >,< Temperature Distribution >,< Wind Aloft >,< Wind Field in Mountainous Terrain >,< Wind Field in Plains >. Now, analyze the relationships between these factors. There are three possible types of relations: Factor A directly affects Factor B Factor B directly affects Factor A No direct effect between two factors Please systematically evaluate the interconnections between the specified factors, focusing only on significant relation

System reply

As a communication and collaboration expert with a background in psychology, sociology, and organizational behavior, I will facilitate effective communication and collaboration among diverse groups to analyze the relationships between the factors associated with the predictive model. We will evaluate the interconnections between the specified factors, focusing only on significant relations.1. Relationship between the scenario and the morning convective inhibition:* Factor A (scenario) directly affects Factor B (morning convective inhibition)2. Level of morning convective inhibition:* Factor A (morning convective inhibition) directly affects Factor B (atmospheric moisture conditions)3. Morning convective inhibition in the given scenario:* Factor A (scenario) directly affects Factor B (morning convective inhibition)4. Atmospheric moisture conditions:* Factor A (atmospheric moisture conditions) directly affects Factor B (morning convective inhibition)5. Relationship between the scenario and the morning atmospheric instability:* Factor A (scenario) directly affects Factor B (morning atmospheric instability)6. Lifted Index (LI) and density relationship:* Factor A (Lifted Index (LI)) directly affects Factor B (density relationship)7. Morning atmospheric instability in the given scenario:* Factor A (scenario) directly affects Factor B (morning atmospheric instability)8. Morning atmospheric instability:* Factor A (morning atmospheric instability) directly affects Factor B (combined vertical motion)9. Combined vertical motion:* Factor A (combined vertical motion) directly affects Factor B (mesoscale area)10. Mesoscale area:* Factor A (mesoscale area) directly affects Factor B (combined moisture conditions)11. Combined moisture conditions:* Factor A (combined moisture conditions) directly affects Factor B (area of dry air)12. Area of dry air:* Factor A (area of dry air) directly affects Factor B (outflow from the mountains)13. Outflow from the mountains:* Factor A (outflow from the mountains) directly affects Factor B (wind hodograph)14. Wind hodograph:* Factor A (wind hodograph) directly affects Factor B (morning boundaries)15. Morning boundaries:* Factor A (morning boundaries) directly affects Factor B (overall boundaries)16. Overall boundaries:* Factor A (overall boundaries) directly affects Factor B (Composite Planetary Forecast)17. Composite Planetary Forecast:* Factor A (Composite Planetary Forecast) directly affects Factor B (Convective Available Potential Energy (CAPE) Change)18. Convective Available Potential Energy (CAPE) Change:* Factor A (Convective Available Potential Energy (CAPE) Change) directly affects Factor B (Convective Available Potential Energy (CAPE) in Scenario)19. Convective Available Potential Energy (CAPE) in Scenario:* Factor A (Convective Available Potential Energy (CAPE) in Scenario) directly affects Factor B (Instability in Mountainous Terrain)20. Instability in Mountainous Terrain:* Factor A (Instability in Mountainous Terrain) directly affects Factor B (Cloud Shade due to Convection)21. Cloud Shade due to Convection:* Factor A (Cloud Shade due to Convection) directly affects Factor B (Combined Clouds)22. Combined Clouds:* Factor A (Combined Clouds) directly affects Factor B (Other Cloud Shade)23. Other Cloud Shade:* Factor A (Other Cloud Shade) directly affects Factor B (Visible Cloud Cover)24. Visible Cloud Cover:* Factor A (Visible Cloud Cover) directly affects Factor B (Infrared Cloud Cover)25.

Infrared Cloud Cover:* Factor A (Infrared Cloud Cover) directly affects Factor B (Saturation Content Moisture)26. Saturation Content Moisture:* Factor A (Saturation Content Moisture) directly affects Factor B (Rao Content Moisture)27. Rao Content Moisture:* Factor A (Rao Content Moisture) directly affects Factor B (0.7 Micrometer Vertical Motion)28.0.7 Micrometer Vertical Motion:* Factor A (0.7 Micrometer Vertical Motion)

User prompt

....

System reply

Here is a valid JSON representation of the relationships between the specified factors:"'json[

["weather Scenario", "level of Morning Convective Inhibition"], ["level of Morning Convective Inhibition", "Atmospheric Moisture Conditions"], ["Atmospheric Moisture Conditions", "Lifted Index (LI) and Density Relationship"], ["Lifted Index (LI) and Density Relationship", "Morning Atmospheric Instability"], ["Morning Atmospheric Instability", "Combined Vertical Motion"], ["Combined Vertical Motion", "Mesoscale Area"], ["Mesoscale Area", "Combined Moisture Conditions"], ["Combined Moisture Conditions", "Area of Dry Air"], ["Area of Dry Air", "Outflow from the Mountains"], ["Outflow from the Mountains", "Wind Hodograph"], ["Wind Hodograph", "Morning Boundaries"], ["Morning Boundaries", "Overall Boundaries"], ["Overall Boundaries", "Composite Planetary Forecast"], ["Composite Planetary Forecast", "Convective Available Potential Energy (CAPE) Change"], ["Convective Available Potential Energy (CAPE) Change", "Convective Available Potential Energy (CAPE) in Scenario"], ["Convective Available Potential Energy (CAPE) in Scenario", "Instability in Mountainous Terrain"], ["Instability in Mountainous Terrain", "Cloud Shade due to Convection"], ["Cloud Shade due to Convection", "Combined Clouds"], ["Combined Clouds", "Other Cloud Shade"], ["Other Cloud Shade", "Visible Cloud Cover"], ["Visible Cloud Cover", "Infrared Cloud Cover"], ["Infrared Cloud Cover", "Saturation Content Moisture"], ["Saturation Content Moisture", "Rao Content Moisture"], ["Rao Content Moisture", "0.7 Micrometer Vertical Motion"], ["0.7 Micrometer Vertical Motion", "Subjective Vertical Motion"], ["Subjective Vertical Motion", "Quasi-Geostrophic Vertical Motion"], ["Quasi-Geostrophic Vertical Motion", "Latest Convective Available Potential Energy (CAPE) Inhibition"], ["Latest Convective Available Potential Energy (CAPE) Inhibition", "Lifted Level Index Weighted"], ["Lifted Level Index Weighted", "Current Propagation of Convection"], ["Current Propagation of Convection", "Date of Weather Forecast"], ["Date of Weather Forecast", "Weather Scenario"], ["Weather Scenario", "Atmospheric Dewpoints"], ["Atmospheric Dewpoints", "Low-Level Moisture Advection"], ["Low-Level Moisture Advection", "Instability Change"], ["Instability Change", "Instability Scale in Scenario"], ["Instability Scale in Scenario", "Low-Level Lapse Rate"], ["Low-Level Lapse Rate", "Mean Relative Humidity"], ["Mean Relative Humidity", "Mid-Level Lapse Rate"], ["Mid-Level Lapse Rate", "Mountainous Forecast"], ["Mountainous Forecast", "Movement Features"], ["Movement Features", "Scenario Relative to 3-4"], ["Scenario Relative to 3-4", "Plains Forecast"], ["Plains Forecast", "North 34-Star Forecast"], ["North 34-Star Forecast", "Scenario Relative Planetary Forecast"], ["Scenario Relative Planetary Forecast", "Region 5 Forecast"], ["Region 5 Forecast", "Relative Humidity Ratio"], ["Relative Humidity Ratio", "Surface Wind Shift Distance"], ["Surface Wind Shift Distance", "Synoptic Forcing"], ["Synoptic Forcing", "Temperature Distribution"], ["Temperature Distribution", "Wind Aloft"], ["Wind Aloft", "Wind Field in Mountainous Terrain"], ["Wind Field in Mountainous Terrain", "Wind Field in Plains"]]"'

F Future work

There are several ways to improve the performance of the described algorithms. For both *Harness* and our method, one may try to apply the algorithm for the decreased number of nodes several times, until all possible pairs of nodes have been located inside one request at least once. For our method, more interaction between different "LLM experts" could be established. Similarly to the practices used in (Xiong et al., 2023; Zhuge et al., 2023) it may be useful to engage different experts in a discussion of their original request, and then independently ask whether they want to change their original decision after the discussion. Finally, it may be interesting to fine-tune LLMs to the specific task of BNSL using the existing BNs.