QUITE: Quantifying Uncertainty in Natural Language Text in Bayesian Reasoning Scenarios

Timo Pierre Schrader[1,2] **Lukas Lange**[1] **Simon Razniewski**[3] **Annemarie Friedrich**[2]
[1]Bosch Center for Artificial Intelligence, Renningen, Germany
[2]University of Augsburg, Augsburg, Germany
[3]ScaDS.AI & TU Dresden, Dresden, Germany
timo.schrader|lukas.lange@de.bosch.com
simon.razniewski@tu-dresden.de
annemarie.friedrich@informatik.uni-augsburg.de

Abstract

Reasoning is key to many decision making processes. It requires consolidating a set of rule-like premises that are often associated with degrees of uncertainty and observations to draw conclusions. In this work, we address both the case where premises are specified as numeric probabilistic rules and situations in which humans state their estimates using words expressing degrees of certainty. Existing probabilistic reasoning datasets simplify the task, e.g., by requiring the model to only rank textual alternatives, by including only binary random variables, or by making use of a limited set of templates that result in less varied text.

In this work, we present QUITE, a question answering dataset of real-world Bayesian reasoning scenarios with categorical random variables and complex relationships. QUITE provides high-quality natural language verbalizations of premises together with evidence statements, and expects the answer to a question in the form of an estimated probability. We conduct an extensive set of experiments, finding that logic-based models outperform out-of-the-box large language models on all reasoning types (causal, evidential, and explaining-away). Our results provide evidence that neuro-symbolic models are a promising direction for improving complex reasoning. We release QUITE and code for training and experiments on Github.[1]

1 Introduction

Reasoning about causality is an integral part of intelligence, as it helps to understand and predict the world. In the real world, causes and associations can rarely be determined with complete certainty, and reasoning becomes inherently difficult if uncertainties are involved (Pearl, 1989). An automated system for interpreting text describing causal relationships and their associated numeric probabilities or verbalized degrees of uncertainty would be highly useful in domains such as requirements engineering (Yang et al., 2012) or text-mining in clinical documentation (Turner et al., 2021). Modeling linguistically expressed uncertainty has been an active research area for decades in natural language processing (NLP) (Szarvas et al., 2008; Jean et al., 2016; Sileo and Moens, 2023).

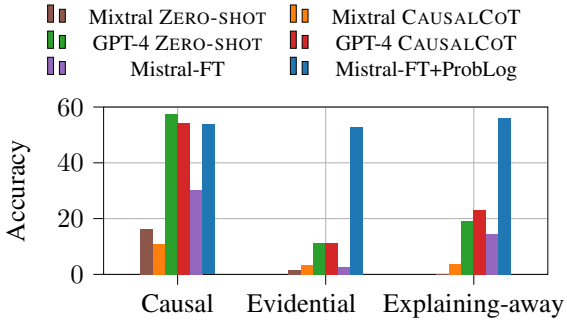
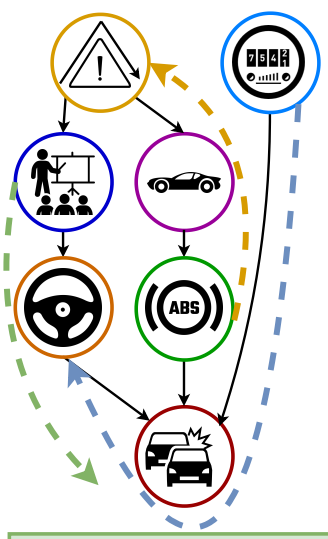


Figure 1: Percentage of instances solved correctly for each **Bayesian reasoning type**. The neuro-symbolic Mistral-FT+ProbLog approach is robust against the inherent difficulties of different reasoning types.

Recently, large language models (LLMs) have shown superior performance on many NLP tasks. However, they fall short of incorporating principled reasoning mechanisms, with frequent failure cases (Kiciman et al., 2023), and their mathematical skills decline if presented with unseen cases (Frieder et al., 2023; Yousefzadeh and Cao, 2023). In zero-shot or chain-of-thought (CoT) prompting settings, open-source and open-weights LLMs are also unable to outperform a random baseline in Bayesian inference in higher-order causal networks (Jin et al., 2023). GPT-3 and GPT-4 are somewhat better, yet do not excel at the task.

To probe LLMs for their reasoning capabilities, Jin et al. (2023) compile the CLADDER dataset based on toy causal inference scenarios taken from textbooks and literature on causal reasoning. CLADDER evaluates performance by *rungs* of the

[1]https://github.com/boschresearch/quite-emnlp24



Numeric Background Premises

In 2% of the cases, the risk aversion behaviour of a car owner can be described as psychopathic, adventurous in 58% of the cases, normal in 30% and cautious in 10% of the cases. [...] For an older luxury model, there's a 30% probability it has anti-lock brakes, and a 70% likelihood it doesn't. [...] If a person has a normal risk aversion, there is a 70% probability that they have not taken a senior driving training or safety course, and a 30% chance that they have taken it. [...] If a car has anti-lock brakes, has been driven for twenty thousand miles, and is driven poorly, there is a 40% chance of no accident, a 30% chance of a mild accident, a 20% chance of a moderate accident, and a 10% chance of a severe accident.

WEP-based Background Premises

If the insurance holder is a senior with a cautious behaviour in terms of avoiding risk, then it is highly likely that they have received senior training, and it is improbable that they have not received such training. [...]

Given that the insurance holder has undergone senior training, the probability of the insurance holder having substandard, normal, or expert driving skills is as about half, unlikely, and highly unlikely, respectively. [...] If the make and model of the car is a luxury car and it is a current year model, then anti-lock brakes are almost certainly present in the car. If the car has anti-lock brakes, the mileage is 50,000, and the driving quality is excellent, then it is almost certain that there will be no accident. [...]

Causal Inference

The driver has attended a senior driving training and the car's mileage is 20,000.
What is the likelihood of being involved in a moderate accident? *p = 0.685*

Evidential Inference

The car has an antilock breaking system.
What is the likelihood that the owner's risk aversion is high? *p = 0.724*

Explaining away

There was a mild accident.
The mileage of the car is 1,000,000.
What is the likelihood for the driver having poor driving skills? *p = 0.1*

Figure 2: Example instances of QUITE. Each question is categorized according to the reasoning pattern.

ladder of causation (Pearl and Mackenzie, 2018).[2] While this work inspired ours, it suffers from several limitations: First, its networks only include binary random variables. Second, questions are of the form "Does X increase the likelihood of Y?" and expect yes/no answers (with a 50:50 distribution). Hence, it is not designed to estimate the correlation of model output with the probabilities estimated according to the Bayesian network.

In this paper, we present QUITE, a new benchmark for **Q**uantifying **U**ncertainty **i**n natural language **Te**xt. As illustrated in Figure 2, QUITE goes one step further, leveraging toy and real-world causal networks and asking the model to output a much finer-grained numeric probability estimate. In addition, our work is the first to make use of categorical random variables (and not just binary variables as in existing related datasets). Despite using real-world networks, our dataset is not solvable from a question-evidence baseline alone, which demonstrates that the model cannot solve the task solely from background knowledge acquired during pre-training. To the best of our knowledge, our work is the first to explicitly distinguish the **three Bayesian inference types** *causal inference*, *evidential reasoning*, and *explaining-away*, which directly reflect the reasoning paths in the network.[3]

Following the recent trend of probing LLMs for their mathematical capabilities, the BLInD dataset (Nafar et al., 2024) focuses on the numeric Bayesian reasoning capabilities of GPT models, hence only uses dummy event variables (e.g., "orange event"). Nafar et al. find that using program-aided language models (Gao et al., 2023) and neuro-symbolic approaches can drastically increase model performance. In their work, however, it remains an open research question whether this approach scales to less template-like and more varied natural language text. QUITE goes one important step into this direction: with the support of LLMs, we verbalize complex real-world Bayesian reasoning scenarios in a linguistically more varied style. Our validation shows that instances in QUITE are of high accuracy with regard to probabilistic information, and use more complex yet mostly error-free language compared to existing datasets.

Existing datasets focus on probing for Bayesian reasoning capabilities when presented with verbalized *numeric* conditional probability tables. QUITE also offers a setting that mimicks human conver-

[2]The rungs of the ladder are: (1) statistical dependencies based on observations: "If I am vaccinated, how likely am I to survive?" (2) interventions: "If I get vaccinated, what is the likelihood of surviving?" (3) counterfactual reasoning: "Would a person have survived if they had been vaccinated?"

[3]We address them in situations corresponding to rung 1 of the causal ladder.

sation, replacing probabilities with *words of estimative probability* (WEPs) such as "unlikely" or "improbable." This scenario has previously only been investigated in the context of natural language inference (Sileo and Moens, 2023). While in this setting, parsing is more difficult for all models, performance is encouraging. Our results illustrate that when targeting natural text, structured causal models in combination with LLMs are a promising approach to estimating likelihood.

A highly interesting finding of our experimental study is that all included LLMs (including GPT models) fail on questions requiring evidential and explaining-away reasoning both in zero-shot and CoT settings (see Figure 1). We hypothesize that the LLMs in our study have learned a good concept of causality during pre-training, and that causal reasoning scenarios can often be solved based on statistical patterns. Potentially, LLMs incorporate biases for assuming causal analyses, as these types of relationships are more frequently expressed in pretraining data. By contrast, our experimental results demonstrate that a fine-tuned neuro-symbolic system has no difficulties solving the latter two categories as well. We hence conclude that for integrating complex reasoning capabilities into NLP systems, neuro-symbolic models are a promising (if not necessary) direction.

Our contributions are as follows: (1) We present a novel dataset of verbalizations of Bayesian networks including categorical variables in two versions (with explicit probabilities vs. words of estimative probability), symbolic target representations, and question-evidence pairs that provide simulated observations and queries asking for probabilities. (2) To the best of our knowledge, all closely related recent prior work uses either vanilla or CoT LLMs, or generates neuro-symbolic representations simply via manually designed prompts. Our work is the first to explicitly fine-tune state-of-the-art LLMs on semantic parsing to probabilistic first-order programming language ProbLog (De Raedt et al., 2007; Fierens et al., 2013). It consistently and strongly outperforms purely LLM-based approaches in probabilistic reasoning on QUITE.

2 Background and Related Work

In this section, we start by introducing the basic concepts of Bayesian networks and corresponding reasoning types, closely following the terms and definitions of Koller and Friedman (2009). We then review existing literature on modeling uncertainty in language, benchmark datasets for Bayesian reasoning, and semantic parsing to logical forms.

Bayesian Networks and Reasoning Patterns. Bayesian networks (BNs) represent joint probability distributions over a set of random variables and probabilistic dependencies between them. These networks are modelled as directed acyclic graphs with nodes and directed edges between the nodes. Nodes represent random variables that can take two or more states. Edges correspond to probabilistic dependencies that are represented in so-called *conditional probability tables* (CPTs). A random variable X_i is an observable attribute that can randomly take two or more disjoint states. For example, the outcome of throwing a coin could be either *head* or *tail*. A Bayesian network therefore represents a joint probability distribution over a set of random variables $\{X_1, \ldots, X_n\}$: $\mathbb{P}(X_1, \ldots, X_n)$. A conditional probability distribution, denoted by edges in the network, is a modification of the joint probability in which a random variable is conditioned on one or more other random variables: $\mathbb{P}(X_i|X_j, \ldots)$. Hence, there is now a dependency between X_i and all its parents in the graph, i.e., the value of the parents directly influences the outcome of X_i.

The combined nature of directed edges allows for different reasoning patterns. *Causal* reasoning requires drawing conclusions about an effect if its cause is observed. Vice versa, reasoning about the cause of an observed effect is called *evidential* reasoning. Finally, drawing conclusions about a cause if an effect and further causes of this effect are observed is called *explaining-away*. For a more in-depth introduction to Bayesian networks and reasoning patterns, please refer to Appendix A.

Modeling uncertainty. BioScope (Szarvas et al., 2008; Farkas et al., 2010; Vincze, 2010) is an early work addressing the modeling of uncertainty in biomedical text by marking triggers and their scope. A cluster of works has focused on modal verbs which are a frequent trigger (Ruppenhofer and Rehbein, 2012; Zhou et al., 2015; Henning et al., 2022; Wagner and Zarrieß, 2023; Owan et al., 2023).

Recently, much research concentrates on probing how LLMs react to prompts containing expressions of (un)certainty. Zhou et al. (2023) find that LLMs are highly sensitive to epistemic markers of certainty in the prompt, decreasing question answering (QA) performance drastically. Conversely,

	QUITE	CLADDER	BLInD
Categorical variables	✓	×	×
Rung of causations	×	✓	×
WEP-based uncertainty	✓	×	×
ProbLog representations	✓	×	✓
Topic/Domain variety	✓	✓	×

Table 1: Comparison of QUITE, CLADDER, and BLInD.

models have been tested with regard to whether they can express their own confidence in an answer (Tian et al., 2023).

Modeling uncertainty has been investigated using Natural Language Inference (NLI) tasks. Sileo and Moens (2023) frame uncertainty-based reasoning as NLI to study how LLMs deal with *words of estimative probability* (WEP) such as "likely" or "improbable." The task of Uncertain NLI (UNLI) (Chen et al., 2020) targets predicting a numeric score for the uncertainty in entailment between two (non-quantified) statements. Talman et al. (2023) explicitly model the variation in judgments of NLI instances exhibited by groups of annotators.

Bayesian reasoning: benchmark datasets. CLADDER (Jin et al., 2023) consists of 10k instances of verbalized Bayesian networks and associated questions. Their *stories* provide an overall summary of the direct effects and spell out the CPTs. The BLInD dataset Nafar et al. (2024) tests to what extent GPT-3.5 and GPT-4 can perform Bayesian reasoning using template-based descriptions of dummy events. In contrast, QUITE focuses on complex real-world scenarios. CLADDER and BLInD instances are verbalized exclusively based on templates, while QUITE exhibits a much larger linguistic variety and higher grammaticality. Key differences between the three datasets are summarized in Table 1. All three studies (including ours) find that basic QA prompting does not work very well, but that COT prompting brings improvements. One example instance from of each dataset is provided in Appendix I.

Semantic parsing to logical form. Constructing structured representations from natural language text has been a long-standing research area in NLP (Zettlemoyer and Collins, 2007; Reddy et al., 2016; Kim et al., 2021). Recent work involves the specialization of LLMs on this task. Olausson et al. (2023) present a framework called *LINC* that translates logical statements into domain-specific languages, where the LLM acts as semantic parser and bridges the gap between natural language and structured, neuro-symbolic representations. Ye et al. (2023) employ an LLM to generate declarative sets of rules that are handed over to a SAT solver executable. Nafar et al. (2024) prompt LLMs to generate symbolic ProbLog code for solving the BLInD dataset.

3 Dataset

In this section, we describe the dataset creation process of QUITE, provide dataset statistics, assess its linguistic quality, and validate the translations of the logical structures into natural language.

3.1 Dataset Structure

QA instances in QUITE are composed from the following parts that facilitate testing model behavior when performing Bayesian inference and reasoning with uncertainty. For an example, see Figure 2.

Numeric background premises are verbalizations of CPTs explicitly mentioning percentages.: *In 2% of the cases, the risk aversion behaviour of a car owner can be described as psychopathic [...]*

WEP-based background premises are verbalizations of CPTs replacing every numeric probability value by an uncertainty quantifier (cf. Section 3.3.1): *There is almost no chance that the risk aversion behaviour of a car owner can be described as psychopathic [...]*

Question-evidence (QE) pairs: Evidences are observations that set the value of one or multiple random variables in the Bayesian model to a particular value. Queries are then asking for the probability of a single random variable that is inferable given the evidence, i.e., $\mathbb{P}(X_Q = x_q^{m_q} | X_{E_1} = x_{E_1}^{m_1}, \ldots, X_{E_j} = x_{E_j}^{m_j})$, where $x_{E_i}^{m_i}$ refers to the value that is assigned to random variable X_{E_i}.

3.2 Data Collection

Our dataset is composed from a collection of publicly available BNs compiled from the literature. They reflect realistic probabilistic relationships in several domains (medicine, severe weather forecasting, car insurance, mildew growth, phytophthora species, protein signalling, water treatment, and software troubleshooting). Our first data source is the *bnlearn* library (Scutari, 2010), which is commonly used in benchmarking scenarios for algorithms for BNs (Liu et al., 2022; Daly and Shen,

2009; Lu et al., 2012). Our second source is the *BNMA BN repository*.[4] Some of the BNs contain node counts in the order of magnitude 100 or 1000, hence, to keep the networks manageable in terms of size, we split them into subnetworks. We marginalize root nodes in subnetworks if they have ancestors in the larger original network to obtain self-contained BNs. An example for this process is provided in Appendix E. We end up with a total of 14 different BNs, split into a total of 30 subnetworks. Our BNs contain nodes of degree 0 to 3, i.e., there are zero to three conditions (parent nodes) on which a random variable can depend. To the best of our knowledge, we are the first to verbalize these widely known networks to natural language, thereby making them available as a resource for NLP research.

3.3 Dataset Creation Steps

We semi-automatically create the natural language part of the dataset with the help of LLMs as illustrated in Appendix H. As LLM backbone in our pipeline, we use Mixtral-8x7B-Instruct-v0.1 (Jiang et al., 2024).[5] Each background premise in the dataset describes the probabilities (either expressed numerically or using WEPs) for all possible assignments to one random variable X_i, given one specific assignment to all the conditions. We generate template-based premises by iterating over every entry of each CPT and fill templates of the form *If [Conditions], then [Probabilities]*, where conditions refer to all incoming edges in the Bayesian network and probabilities to the currently selected variable. Next, to create natural language premises, we prompt Mixtral with a prompt containing technical explanations of the variables in the network, few-shot examples as well as the template-based premises. In contrast to related datasets that fully rely on rule-based templates, QUITE hence contains more varied descriptions.For each network, we create a representation in ProbLog in a semi-automatic way: We manually define predicates for all nodes and categories in the CPTs, and then use a rule-based conversion. For the entire ProbLog data (1192 statements), the first author of the paper has manually checked if the statements match their ProbLog counterparts and if the wording and use of domain-specific vocabulary is consistent throughout the entire verbalized network. This was an extensive manual effort of multiple months. The ProbLog representation enables us to perform fine-tuning of the semantic parser.

3.3.1 WEP-based Background Premises

To guide the LLM to express natural language uncertainty in a principled way, we rely on a human study conducted by Fagen-Ulmschneider (2015) that includes the subjective judgements by more than 100 people who were asked to judge which numeric probability they associate with each adverb in a list. These adverbs are often referred to as *words of estimative probability* (WEP), a term that mainly originates from the work of Kent (1964), which investigates the mapping between specific uncertainty quantifiers and probabilities (see Appendix B). We map each probability value to the closest adverb. In case there is more than one possible adverb (e.g., 10% maps to *improbable*, *little chance*, and *chances are slight*), one of them is randomly selected. Additionally, to simulate subjectivity, we select the second-closest adverb in 10% of the cases. If all states of a random variable have the same probability, we manually correct the verbalization to "equally likely."

The heuristic of choosing the WEPs based on the premises' probabilities works well in most cases, yet we observe that this heuristic does not fully fit cases where all states of a categorical random variable have a low probability. For example, assume we have $\mathbb{P}(X_i = x_i^1) = 0.2$, $\mathbb{P}(X_i = x_i^2) = 0.2$, $\mathbb{P}(X_i = x_i^3) = 0.3$, and $\mathbb{P}(X_i = x_i^4) = 0.3$. This would lead to the following verbalization: *It is probably not the case that X_i takes the value x_i^3 or x_i^4, and it is unlikely that it takes x_i^1 or x_i^2*. We manually add additional information to these instance describing the state that is still the most likely one. We leave the adaption of WEPs to this edge case as a direction for future research.

3.3.2 Question-Evidence (QE) Pairs

We construct QE pairs as follows. As evidences, we randomly sample 1 to n-1 observations per instance (i.e., $X_{E_1} = x_{E_1}^{m_1}, \ldots, X_{E_j} = x_{E_j}^{m_j}$) and let Mixtral transform them to natural language statements such as "The accident was mild." For the question, we sample one node X_q for which Mixtral formulates a question of the form: "What is the likelihood of X_q having the value x_Q^m?" Each QE pair requires calculating the probability

[4] https://www.abnms.org/bnrepo/

[5] We also experiment with Mixtral-8x22B-Instruct-v0.1, but due to budget reasons, we mostly stick to Mixtral-8x7B-Instruct-v0.1. A small percentage (ca. 8%) of the dataset has been created manually.

	# Train	# Test	Total
Networks	20	10	30
Numeric premises	930	273	1192
WEP-based premises	930	273	1192
Evidence statements	812	578	1390
Queries	347	230	577
Avg. # states / rand. var.	$3.5_{\pm 2.0}$	$2.9_{\pm 1.4}$	$3.3_{\pm 1.9}$
Avg. # rand. var. / net	$5.9_{\pm 2.4}$	$6.0_{\pm 2.5}$	$5.9_{\pm 2.4}$
Avg. # premises / net	$46.5_{\pm 31.2}$	$27.3_{\pm 23.2}$	$40.1_{\pm 30.2}$

Table 2: Dataset statistics for QUITE. Subscripts denote standard deviation.

$\mathbb{P}(X_Q = x_Q^m | X_{E_1} = x_{E_1}^{m_1}, \ldots, X_{E_j} = x_{E_j}^{m_j})$. The ground truth answer (numeric probability value) is calculated based on the underlying probabilistic model of each subnetwork. Most QE instances can be clearly categorized into their respective reasoning pattern. We determine *causal* and *evidential* reasoning by inspecting the list of parent and child nodes, respectively, and check if one of them is observed, i.e., part of the evidence. To identify explaining-away QEs, we only check if one of the direct child nodes of X_Q and one of their direct parents is observed. The test set contains 92 causal, 62 evidential and 26 explaining-away QE instances.

3.4 Dataset Statistics

We provide detailed statistics for QUITE in Table 2. We ensure that subnetworks that are derived from the same original network are assigned to only training or test data, respectively. All QE pairs have been manually checked and if necessary corrected by the first author. On average, there are three to four states per random variable. The average number of background premises reflects the amounts of probabilistic statements that need to be processed before reasoning. The average number of premises per network in QUITE is much higher than those in related works, reflecting its challenging nature. The statistics in the lower part of Table 2 differ between training and test split due to taking the original networks into account.

3.5 Validation and Quality Assessment

In contrast to QE pairs, premise statements are assembled in a semi-automatic fashion. In this section, we validate their correctness and examine the linguistic quality of the entire dataset.

Validation. The first author of the paper has performed extensive checking and correcting for the 2384 premise statements. As a second validation step, two of the (non-first) authors of this paper that were not exposed to the generation process before are presented with 400 randomly sampled premises of QUITE (200 numeric and 200 WEP-based premises). They are asked to assess whether the LLM-generated output contains all input variables and states and whether probabilities have been translated correctly. Of the numeric premises, 193 instances (96.5%) correctly describe the underlying probability distribution without ambiguities. Most errors relate to rounding close-to-zero probabilities to zero. Of the WEP-based instances, 188 instances (94%) are correct, with the LLM misinterpreting the input and wrong representations of the probability values being the main error causes. Our validation study shows that QUITE contains mostly well-formed instances that correctly reflect the random variable states and probabilities.

Linguistic Quality Assessment. To assess the linguistic quality of QUITE, we make use of Grammarly,[6] a state-of-the-art commercial writing assistant. We compare QUITE to CLADDER (excluding its non-sensical subset) and BLInD. We randomly sample premises, evidences, and queries until a character count of approximately 95,000 has been reached.

Results are provided in Table 3. On average, QUITE has much fewer grammar and spelling mistakes per instance than CLADDER.[7] This highlights the advantage of LLM-based generation over template-based instance generation. QUITE has the most specific vocabulary, demonstrated by the highest amount of rare words, which Grammarly defines as words that do not belong to the 5k most frequent English words. According to the Flesh-Kincaid readability score (Kincaid, 1975), BLInD requires skills of the level of 8th/9th graders, CLADDER and the WEP-based part of QUITE need 10th to 12th grade skills, and the numeric part of QUITE requires college-level reading skills.

Finally, QUITE edges out on the two other datasets in terms of the overall Grammarly score. This as a strong indicator that our dataset is much closer to human-like natural language. We conclude from this analysis that our dataset makes use of rich language with a complex vocabulary, and is close to human-like language. Overall, QUITE is well-suited for assessing the reasoning capabilities of state-of-the-art models in realistic scenarios.

[6] https://app.grammarly.com/

[7] BLInD templates lack determiners: "If purple event is False, then grey event is True with probability of 39%."

Grammarly Metric	QUITE		CLAD.	BLInD
	Num.	WEP		
Writing Issues / Instance	0.3	0.3	1.3	30.9
Rare words	43%	41%	36%	23%
Readability	48	50	50	68
Overall judgement (0-100)	**85**	**82**	45	34

Table 3: Dataset quality assessment by Grammarly for a randomly sampled subset of all datasets.

4 Modeling

In this section, we describe the various models that we benchmark using QUITE. Fine-tuned models are suffixed *-FT* in the following.

4.1 LLM Prompting Methods

We experiment with several prompting techniques for state-of-the-art LLMs of different sizes.[8] In the **zero-shot** setting, we provide all background premises of the network, a set of evidences and a question asking for the probability of a specific random variable taking a selected state.[9] The **CAUSALCOT** technique was introduced by Jin et al. (2023) and asks the model to build up the probabilistic graph, to extract the question type and to perform the mathematical calculation step by step.

4.2 Neuro-symbolic Approach

Our ProbLog-based approach separates problem understanding and probabilistic reasoning, first parsing each premise (both numeric and WEP-based), evidence statements, and queries into a ProbLog program. In logic programming languages, declarative programs are defined as a series of rules and facts that, in combination, evaluate to true or false. In Prolog, rules are defined in first-order logic, where a rule body defines which conditions need to be met (i.e., need to be *true*) in order for the rule head to be evaluated as true. ProbLog (De Raedt et al., 2007; Fierens et al., 2013), which we use in this work, is a **probabilistic programming language** that extends the functionality of Prolog. It allows the specification of probabilistic models by declaring the probability distributions in FOL-style formulas. Since our dataset comprises not only binary, but also categorical random variables, we use annotated disjunctions, e.g.,

```
0.02::risk_aversion(car_owner, psychopathic);
0.58::risk_aversion(car_owner, adventurous);
0.30::risk_aversion(car_owner, normal);
0.10::risk_aversion(car_owner, cautious).
```

We first parse background premises into ProbLog using either a zero-shot LLM (**ProbLog-Prompt**) or an LLM fine-tuned for text-to-ProbLog parsing (**ProbLog-FT**). The QE pairs are parsed as a second step (i.e., after the vocabulary of predicates has been determined by the premise parsing step). This is in particular important when using a prompt-based LLM for semantic parsing. After parsing into a ProbLog program, the solver executes the code to determine the answer. A full ProbLog example for QUITE is provided in Appendix D. For ProbLog-FT, we use Mistral-7B due to its large context window of 32k tokens.

4.3 Baselines

As a trivial baseline, we report a system always predicting 50%. The **Regression-FT** is a Llama2-7B model (Touvron et al., 2023) trained for regression with sigmoid output given all premises, evidences and the question. The input to the regression layer is the embedding of the last token. Additionally, we fine-tune a Mistral-7B model on predicting the probability as text, e.g., "The probability is p" (**LLM-FT**).

5 Experimental Evaluation

In this section, we report our extensive evaluation of current state-of-the-art LLMs and our neurosymbolic model on QUITE. All fine-tuning details and parameters are provided in Appendix C.

5.1 Evaluation Metrics

As QUITE comprises two versions of the background premises, we can investigate the following research questions: (1) Given numeric premises, evidences, and a question, can the model(s) correctly calculate the likelihood of events/states? (2) In the case of linguistically specified uncertainty (WEP-based premises), can the model provide close estimates of the likelihood of events/states? For each model, we hence report the percentages of **correct** predictions,[10] **wrong** predictions, and **error** cases without a valid numeric answer (e.g., in case of invalid ProgLog programs or if an LLM refuses

[8] GPT4-Turbo (*turbo-2024-04-09*) (OpenAI, 2024), Llama-3-8B-Instruct (AI@Meta, 2024), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023), and Mixtral-8x7B-Instruct-v0.1 (Jiang et al., 2024). The temperature is set to 0.0.

[9] We also performed preliminary experiments with an one-shot example which did not result in consistent improvements.

[10] We evaluate with a relative tolerance of 10^{-4} to compensate floating point errors.

to answer). RSME, computed as $\sqrt{\sum_{i=1}^{n} \frac{(p_i - \hat{p_i})^2}{n}}$ reports how far numeric estimates deviate from the ground truth. We report two variations for handling error cases: $\text{RMSE}_{50\%}$ has a fallback to 50% as default answer for any invalid model output. The rationale behind choosing 50% as fallback value for $\text{RMSE}_{50\%}$ is that whenever a model refuses to answer or produces invalid output (e.g., erroneous Problog code), we can only make a random guess. Since 50% reflects an equal likelihood of something being the case or not, it is a natural choice for the fallback case.

$\text{RMSE}_{\text{nonError}}$ is computed only over valid, but not necessarily correct predictions. Note that $\text{RMSE}_{\text{nonError}}$ scores are not comparable across rows. $\text{RMSE}_{50\%}$ does not directly report the quality of *valid* predictions, i.e., where the model or Problog solver return actual numeric values. To also judge the quality of the valid numeric predictions, $\text{RMSE}_{\text{nonError}}$ only takes instances into account for which there are valid predictions. Since the amount of valid predictions heavily alters between the different models and approaches, this score does not necessarily refer to the same instances between the different models, but can be used to interpret how close the numeric output of a model is to the correct answer.

5.2 Results for Numeric Premises

The results for numeric background premises are reported in the upper part of Table 4. The QE-only baseline represents GPT-4 performance when omitting the premises. Only 3.1% of the cases are solvable by GPT-4 without referring to information in the premises, which means that background knowledge and reasoning is required for solving QUITE instances. This experiment validates the suitability of QUITE to test for probabilistic reasoning performance.

The Regression-FT model does not predict any instances correctly and has a similar performance as the baseline that always predicts the average probability value. LLM-FT correctly predicts the output in 1/5 of the cases without invalid responses, which implies that this method leveraged pre-training information in a better way.

Next, we compare results for two prompting techniques. In terms of accuracy and RMSE, GPT-4 outperforms the open-weights models by a large margin. Jin et al. (2023) report considerable performance gains (10-20%) over zero-shot settings when using CAUSALCOT on CLADDER. On QUITE, we observe that both GPT-4 and Llama-3 slightly profit from the CAUSALCOT technique, but the gain is not as large as observed on CLADDER. Results are somewhat inconclusive as performance for Mixtral even drops when integrating CAUSALCOT.

ProbLog-FT outperforms all other models and approaches by a large margin. It is the only approach that finds the correct answer to about every second question. Outsourcing all mathematical steps that are required to obtain the final answer to an external solver is much more effective overall, also achieving the best RMSE_{50} score, demonstrating a clear benefit over the trivial baselines and over all prompting-based approaches. To substantiate the need for fine-tuning, we prompt GPT-4 on the task of generating ProbLog code (ProbLog-Prompt). This model suffers from producing a very high number of parsing errors (approximately 76% of the cases).

5.3 Results for WEP-based Premises

In the case of WEP-based background premises (lower part of Table 4), we focus on the RMSE scores. Interestingly, when provided with the WEP-based premises, GPT-4 still arrives at the correct solution in 8.7% of all cases, indicating that it leverages the textual descriptions in the premises. Compared to using numeric premises, Problog-FT produces more parsing errors, indicating that more training data is necessary in this setting. Most notably, however, ProbLog-FT (7B parameters) performs on par with GPT-4 (estimated 8x222B parameters), which indicates that fine-tuning neuro-symbolic models is a promising direction to improve automatic reasoning.

5.4 Results by Reasoning Type

Figure 1 (on page 1) breaks down the results by reasoning type, showing how many instances in each category have been solved correctly by the best-performing models. Causal reasoning seems to be the easiest type of reasoning. All models except for ProbLog-FT fail on *evidential* and *explaining-away* reasoning. We conclude that LLMs show reasonable skills in forward-style reasoning, whereas backward-style reasoning seems to be a major issue. Once a valid representation of the causal structure has been assembled, however, our neuro-symbolic models can perform any type of reasoning, illustrating an important advantage of our neuro-symbolic approach.

	Method	Model	Response Metrics			RMSE ↓	
			% correct ↑	% wrong ↓	% error↓	50%	nonError
	50% *baseline*	-	0.9	99.1	0.0	0.363	0.363
	QE only baseline	GPT4-Turbo	3.1	96.9	0.0	0.361	0.361
Numeric premises	ZERO-SHOT	GPT4-Turbo	36.7	57.6	5.7	0.304	0.302
		Llama-3-8B	7.0	91.7	1.3	0.521	0.514
		Mixtral-8x7B	9.6	77.3	13.1	0.441	0.449
	CAUSALCOT	GPT4-Turbo	37.1	61.6	1.3	0.326	0.326
		Llama-3-8B	11.4	87.8	**0.9**	0.403	0.404
		Mixtral-8x7B	7.0	83.0	10.0	0.486	0.498
	Regression-FT	Llama-2-7B	$0.0_{\pm 0.0}$	$100.0_{\pm 0.0}$	$0.0_{\pm 0.0}$	$0.369_{\pm 0.00}$	$0.369_{\pm 0.00}$
	LLM-FT	Mistral-7B	$21.5_{\pm 1.4}$	$78.5_{\pm 1.4}$	$0.0_{\pm 0.0}$	$0.327_{\pm 0.01}$	$0.327_{\pm 0.01}$
	ProbLog-Prompt	GPT4-Turbo	19.2	**4.8**	76.1	0.313	**0.116**
	ProbLog-FT	Mistral-7B	**54.5**$_{\pm 4.8}$	$16.9_{\pm 5.1}$	$28.6_{\pm 7.1}$	**0.244**$_{\pm 0.03}$	$0.179_{\pm 0.05}$
WEP-based premises	ZERO-SHOT	GPT4-Turbo	5.7	82.1	12.2	0.391	0.394
		Llama-3-8B	2.2	83.4	14.4	0.493	0.512
		Mixtral-8x7B	3.5	50.7	45.9	0.484	0.562
	CAUSALCOT	GPT4-Turbo	8.7	89.1	2.2	0.341	0.341
		Llama-3-8B	3.5	91.7	4.8	0.436	0.438
		Mixtral-8x7B	2.6	59.4	38.0	0.456	0.511
	Regression-FT	Llama-2-7B	$0.0_{\pm 0.0}$	$100.0_{\pm 0.0}$	$0.0_{\pm 0.0}$	$0.425_{\pm 0.06}$	$0.425_{\pm 0.06}$
	LLM-FT	Mistral-7B	$3.6_{\pm 0.9}$	$96.4_{\pm 0.9}$	$0.0_{\pm 0.0}$	$0.344_{\pm 0.01}$	$0.344_{\pm 0.01}$
	ProbLog-Prompt	GPT4-Turbo	0.4	8.7	90.8	0.362	**0.268**
	ProbLog-FT	Mistral-7B	$1.3_{\pm 0.6}$	$32.8_{\pm 4.6}$	$65.9_{\pm 4.8}$	**0.319**$_{\pm 0.01}$	$0.299_{\pm 0.04}$
	ProbLog-FT oracle	Mistral-7B	$87.3_{\pm 3.0}$	$5.8_{\pm 1.7}$	$6.9_{\pm 1.6}$	$0.165_{\pm 0.04}$	$0.145_{\pm 0.04}$

Table 4: Results on QUITE for numeric and words of estimative probability (WEP)-based background premises.

5.5 Error Analysis

In this section, we provide qualitative and quantitative analyses for different failure cases of our approaches. Further analyses on the effect of network size on performance are provided in Appendix G.

Neuro-symbolic Approach. The ProbLog-FT model has two main sources of errors: syntax errors and unknown clauses, i.e., using undefined predicates. To get a sense of whether the main source of errors is step 1 (premise parsing) or step 2 (QE parsing), we conduct an oracle experiment (cf. bottom row in Table 4) in which the already parsed premises are provided to the network and only QE parsing is performed by the model. In this setup, the model is able to get four out of five cases right, which strongly indicates that parsing the lengthy premises is the main source of errors.

Prompting. Our qualitative analysis reveals that for prompt-based LLM approaches, **mathematical errors** are a frequent error case, with wrong answers being produced due to rounding errors or erroneous calculations. In other cases, the LLMs refuse to answer because of the mathematical complexity or asks whether it should continue. Occasionally, the models insist on not having enough information to solve the question and conclude the generation without results.

6 Conclusion and Outlook

In this paper, we have presented QUITE, a new question answering dataset that provides Bayesian reasoning scenarios for a variety of domains and that can be used to assess uncertainty-based reasoning with LLMs. From a large set of experiments using numeric probabilistic premises and premises expressed using words of estimative probability, we conclude that a neuro-symbolic approach combining probabilistic logic programming and fine-tuned LLMs as semantic parsers is most promising. Moreover, we find that non-specialized LLMs mostly fail on this task.

Outlook. For increasing the robustness of logic-based semantic parsing models, different approaches should be further investigated. For example, constrained decoding techniques could be used to ensure that only valid predicates can be generated at any point in time. Next steps include also studying reasoning in modal and counterfactual scenarios.

Limitations

In this work, we investigate probabilistic reasoning with uncertainty using verbalized Bayesian networks. This approach assumes that the entire probability distribution is known and given at any time. However, in real-life scenarios (e.g., in data that is obtained from production plants), probability distributions are often underspecified or influencing factors are not even known, i.e., there are hidden variables that influence the relationships. Furthermore, our models act upon a limited number of sentences at once, whereas uncertainty descriptions could also be provided in longer texts that also contain information that is not relevant to the reasoning process.

In its current version, QUITE operates on *rung 1* of the *ladder of causation* (Pearl and Mackenzie, 2018). Our work could of course be extended in the future to also cover rung 2 of the *ladder of causation*, meaning that based on the networks, one could perform interventional queries (*do-operator*) that dynamically modify the probabilistic relationships. To do that, we need to generate additional queries of form *If we force [...], does that lead to [...]* and map that onto the *do/1* predicate in ProbLog. As a first step, however, we decided to carefully study rung-1 questions with regard to three Bayesian reasoning types.

As with most benchmarks these days, there is a potential issue of data contamination, i.e., the LLMs could have seen relevant parts of QUITE in their pre-training corpus. Our natural language corpus is based on plain probability tables. These tables could have been part of the pretraining corpus. Some of the networks in our dataset were described in published work before. Therefore, it could be that some of the relationships between random variables are vaguely known to the LLMs. However, we argue that no paper describes large BNs in every detail, preventing the LLMs from learning every network detail by hard. This assumption is supported by the poor performance of the question-evidence only baseline.

Ethical Considerations

QUITE builds upon data from many different scientific and non-scientific domains. These include different Bayesian networks from domains related to medical treatment and health issues. However, we emphasize that QUITE and our proposed models in their current version should not be used for any kind of reliable decision making in medicine and health-related issues. All probabilistic networks in QUITE only reflect a subset of the entire causal relationships that might exist and are used for assessing self-contained Bayesian reasoning without considering the much broader scientific knowledge available to the world. Furthermore, we did not verify the correctness of the observed data by checking the biomedical literature.

Acknowledgements

We would like to express our deepest gratitude to Marco Scutari, who is the author of *bnlearn* and gave us the permission to use the Bayesian networks for our research. We also thank the anonymous reviewers for their valuable feedback. Finally, we would like to thank our colleagues at Bosch Research for all the valuable discussions, feedback and ideas on and for our work.

References

AI@Meta. 2024. Llama 3 model card.

Tongfei Chen, Zhengping Jiang, Adam Poliak, Keisuke Sakaguchi, and Benjamin Van Durme. 2020. Uncertain natural language inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8772–8779, Online. Association for Computational Linguistics.

Rónán Daly and Qiang Shen. 2009. Learning bayesian network equivalence classes with ant colony optimization. *J. Artif. Int. Res.*, 35(1):391–447.

Luc De Raedt, Angelika Kimmig, and Hannu Toivonen. 2007. Problog: a probabilistic prolog and its application in link discovery. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, page 2468–2473, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Wade Fagen-Ulmschneider. 2015. Perception of probability words.

Richárd Farkas, Veronika Vincze, György Móra, János Csirik, and György Szarvas. 2010. The CoNLL-2010 shared task: Learning to detect hedges and their scope in natural language text. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning – Shared Task*, pages 1–12, Uppsala, Sweden. Association for Computational Linguistics.

Daan Fierens, Guy Van den Broeck, Joris Renkens, Dimitar Sht. Shterionov, Bernd Gutmann, Ingo Thon, Gerda Janssens, and Luc De Raedt. 2013. Inference and learning in probabilistic logic programs using weighted boolean formulas. *CoRR*, abs/1304.6810.

Simon Frieder, Luca Pinchetti, Alexis Chevalier, Ryan-Rhys Griffiths, Tommaso Salvatori, Thomas Lukasiewicz, Philipp Christian Petersen, and Julius Berner. 2023. Mathematical capabilities of chatgpt.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. PAL: program-aided language models. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 10764–10799. PMLR.

Sophie Henning, Nicole Macher, Stefan Grünewald, and Annemarie Friedrich. 2022. MiST: a large-scale annotated resource and neural models for functions of modal verbs in English scientific text. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1305–1324, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Pierre-Antoine Jean, Sébastien Harispe, Sylvie Ranwez, Patrice Bellot, and Jacky Montmain. 2016. Uncertainty detection in natural language: a probabilistic model. In *Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics, WIMS 2016, Nîmes, France, June 13-15, 2016*, pages 10:1–10:10. ACM.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts.

Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, LYU Zhiheng, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, et al. 2023. Cladder: Assessing causal reasoning in language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Sherman Kent. 1964. Words of estimative probability. *Studies in intelligence*, 8(4):49–65.

Emre Kiciman, Robert Ness, Amit Sharma, and Chenhao Tan. 2023. Causal reasoning and large language models: Opening a new frontier for causality. *CoRR*, abs/2305.00050.

Gene Kim, Viet Duong, Xin Lu, and Lenhart Schubert. 2021. A transition-based parser for unscoped episodic logical forms. In *Proceedings of the 14th International Conference on Computational Semantics (IWCS)*, pages 184–201, Groningen, The Netherlands (online). Association for Computational Linguistics.

J.P. Kincaid. 1975. *Derivation of New Readability Formulas: (automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel*. Research Branch report. Chief of Naval Technical Training, Naval Air Station Memphis.

D. Koller and N. Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques*. Adaptive Computation and Machine Learning series. MIT Press.

Yang Liu, Anthony C. Constantinou, and Zhigao Guo. 2022. Improving bayesian network structure learning in the presence of measurement error. *J. Mach. Learn. Res.*, 23(1).

Yang Lu, Mengying Wang, Menglu Li, Qili Zhu, and Bo Yuan. 2012. LSBN: A large-scale bayesian structure learning framework for model averaging. *CoRR*, abs/1210.5135.

Aliakbar Nafar, Kristen Brent Venable, and Parisa Kordjamshidi. 2024. Probabilistic reasoning in generative large language models.

Theo Olausson, Alex Gu, Ben Lipkin, Cedegao Zhang, Armando Solar-Lezama, Joshua Tenenbaum, and Roger Levy. 2023. LINC: A neurosymbolic approach for logical reasoning by combining language models with first-order logic provers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5153–5176, Singapore. Association for Computational Linguistics.

OpenAI. 2024. Gpt-4 technical report.

Risako Owan, Maria Gini, and Dongyeop Kang. 2023. Quirk or palmer: A comparative study of modal verb frameworks with annotated datasets. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pages 183–199, Singapore. Association for Computational Linguistics.

Judea Pearl. 1989. *Probabilistic reasoning in intelligent systems - networks of plausible inference*. Morgan Kaufmann series in representation and reasoning. Morgan Kaufmann.

Judea Pearl and Dana Mackenzie. 2018. *The Book of Why: The New Science of Cause and Effect*, 1st edition. Basic Books, Inc., USA.

Siva Reddy, Oscar Täckström, Michael Collins, Tom Kwiatkowski, Dipanjan Das, Mark Steedman, and Mirella Lapata. 2016. Transforming dependency structures to logical forms for semantic parsing. *Transactions of the Association for Computational Linguistics*, 4:127–140.

Josef Ruppenhofer and Ines Rehbein. 2012. Yes we can!? annotating English modal verbs. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1538–1545, Istanbul, Turkey. European Language Resources Association (ELRA).

Marco Scutari. 2010. Learning bayesian networks with the bnlearn R package. *Journal of Statistical Software*, 35(3):1–22.

Damien Sileo and Marie-Francine Moens. 2023. Probing neural language models for understanding of words of estimative probability.

György Szarvas, Veronika Vincze, Richárd Farkas, and János Csirik. 2008. The BioScope corpus: annotation for negation, uncertainty and their scope in biomedical texts. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, pages 38–45, Columbus, Ohio. Association for Computational Linguistics.

Aarne Talman, Hande Celikkanat, Sami Virpioja, Markus Heinonen, and Jörg Tiedemann. 2023. Uncertainty-aware natural language inference with stochastic weight averaging. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 358–365, Tórshavn, Faroe Islands. University of Tartu Library.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, Singapore. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Shengyi Huang, Kashif Rasul, Alexander M. Rush, and Thomas Wolf. 2023. The alignment handbook. `https://github.com/huggingface/alignment-handbook`.

Mark Turner, Julia Ive, and Sumithra Velupillai. 2021. Linguistic uncertainty in clinical NLP: A taxonomy, dataset and approach. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 12th International Conference of the CLEF Association, CLEF 2021, Virtual Event, September 21-24, 2021, Proceedings*, volume 12880 of *Lecture Notes in Computer Science*, pages 129–141. Springer.

Veronika Vincze. 2010. Speculation and negation annotation in natural language texts: what the case of BioScope might (not) reveal. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pages 28–31, Uppsala, Sweden. University of Antwerp.

Jonas Wagner and Sina Zarrieß. 2023. Probing BERT's ability to encode sentence modality and modal verb sense across varieties of English. In *Proceedings of the 15th International Conference on Computational Semantics*, pages 28–38, Nancy, France. Association for Computational Linguistics.

Hui Yang, Anne N. De Roeck, Vincenzo Gervasi, Alistair Willis, and Bashar Nuseibeh. 2012. Speculative requirements: Automatic detection of uncertainty in natural language requirements. In *2012 20th IEEE International Requirements Engineering Conference (RE), Chicago, IL, USA, September 24-28, 2012*, pages 11–20. IEEE Computer Society.

Xi Ye, Qiaochu Chen, Isil Dillig, and Greg Durrett. 2023. Satlm: Satisfiability-aided language models using declarative prompting.

Roozbeh Yousefzadeh and Xuenan Cao. 2023. Large language models' understanding of math: Source criticism and extrapolation.

Luke Zettlemoyer and Michael Collins. 2007. Online learning of relaxed CCG grammars for parsing to logical form. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 678–687, Prague, Czech Republic. Association for Computational Linguistics.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.

Kaitlyn Zhou, Dan Jurafsky, and Tatsunori Hashimoto. 2023. Navigating the grey area: How expressions of uncertainty and overconfidence affect language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5506–5524, Singapore. Association for Computational Linguistics.

Mengfei Zhou, Anette Frank, Annemarie Friedrich, and Alexis Palmer. 2015. Semantically enriched models for modal sense classification. In *Proceedings of the First Workshop on Linking Computational Models of Lexical, Sentential and Discourse-level Semantics*, pages 44–53, Lisbon, Portugal. Association for Computational Linguistics.

A Theoretical Background

This section introduces the main theoretical concepts on probabilistic reasoning. We closely follow the notations and definitions of Koller and Friedman (2009).

A.1 Bayesian Networks

A Bayesian network is a directed acyclic graph (DAG) $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ that represents a joint probability distribution $\mathbb{P}$ over a set of random variables $\{X_1, \ldots, X_n\}$. Each random variable X_i can take values from their respective domain Ω_i, which is the set of possible realizations $\{x_i^1, \ldots, x_i^k\}$, where $k = |\Omega_i|$. If $k = 2$, then we say that X_i adheres to a *Bernoulli* distribution, whereas $k > 2$ makes them a *categorical* random variable. Furthermore, to ensure a valid probability distribution, it must always hold that $\sum_{j=1}^{k} \mathbb{P}(X_i = x_i^j) = 1$.

Each node $v_i \in \mathbf{V}$ in $\mathcal{G}$ represents one variable X_i. An edge $e_{i,j} \in \mathbf{E}$ between two nodes v_i, v_j represents a correlation between the two associated random variables and thereby models the conditional probability distribution (CPD) $\mathbb{P}(X_j|X_i)$. These CPDs are of special importance when dealing with so-called *observations*. Mathematically speaking, observations modify the joint probability distribution over $\mathcal{G}$ as follows:

$$\mathbb{P}(X_1, \ldots, X_{j-1}, X_{j+1}, \ldots, X_n | X_j = x_j^m) = \frac{\mathbb{P}(X_1, \ldots, X_j = x_j^m \ldots, X_n)}{\mathbb{P}(X_j = x_j^m)}$$

This means, to obtain the value of the full probability distribution, only one state of X_j must be considered instead of the whole range of possible states.

A.2 Reasoning Patterns

Directed edges in a Bayesian network not only indicate a dependence, but they also allow for different types of reasoning when observing one or more random variables. In the following, we assume a simple three-node network with nodes X_1, X_2, X_3 and edges $X_1 \rightarrow X_3$ and $X_2 \rightarrow X_3$.

Causal Reasoning: We know the cause and draw conclusions about the effect. Suppose we look at the connection $X_1 \rightarrow X_3$ and observe the value of X_1. This gives us a strong hint on what the status of X_3 could likely be. The underlying probability distribution is $\mathbb{P}(X_3|X_1)$. For instance, assume we observe that the weather is

rainy. This leads us to the conclusion that the streets are very likely to be wet.

Evidential Reasoning: The other way round is to observe X_3 and reason about X_1 in $X_1 \rightarrow X_3$. We now know about the value of the effect, which we can use to make assumptions about the most likely cause. The mathematical term is $\mathbb{P}(X_1|X_3)$. This probability is not directly represented in this Bayesian network, but it can be calculated by using Bayes' theorem: $\mathbb{P}(X_1|X_3) = \frac{\mathbb{P}(X_1,X_3)}{\mathbb{P}(X_3)}$. Let us now assume we observe wet streets. This gives us strong hints on whether it has been raining before.

Explaining-Away: This requires multiple causes with a common effect, shaping a so-called *v-structure* ($X_1 \rightarrow X_3 \leftarrow X_2$). Now observing one of the potential causes (X_1 or X_2) and the effect "explains the influence of the other causes away." Assume that we again observe wet streets. This could be due to rain, but also due road cleaning machines. If we now obtain the knowledge that it is raining or was raining, we can assume that road cleaning is unlikely. The state of the weather "explains away" the need for road cleaning machines.

We enrich our dataset by categorizing the queries into their respective reasoning type(s) if applicable, making it possible to also investigate the robustness of LLMs with respect to different reasoning patterns.

B Words of Estimative Probability (WEP)

Table 5 lists the WEPs which are used to model uncertainty in QUITE. The table provides a mapping between adverbs and numeric probabilities, estimated via a survey conducted by Fagen-Ulmschneider (2015). Every numeric value is mapped to the closest adverb, not considering the confidence intervals in the table. However, we introduce one exception in the mapping: if a numeric probability is below 45%, it is not mapped to the closest adverb (i.e., *about even*), but instead to *probably not*. This is to make sure that values of 38% for instance are not mapped to *about even*.

C Fine-Tuning Details

We fine-tune different modelling approaches on QUITE as described in Section 4 to see if specialized models can outperform out-of-the-box baseline models. Since state-of-the-art models are made up of billions of parameters, we use LoRA (Hu et al., 2021) to fine-tune low-rank adapters on QUITE. We use the code from Tunstall et al. (2023) to fine-tune all models on Nvidia H100 and A100 GPUs. All models are trained using full precision, i.e., FP32 to make sure that we do not lose performance. The hyperparameters for each model are listed in Table 6. To keep fine-tuning sustainable, we refrained from performing an extensive hyperparameter search. Instead, we selected commonly used values (e.g., cf. Zheng et al. (2024)). Models are selected based on their performance on the development set, which is a subset of train, and evaluated only once on test.

WEP	Associated Prob.
certain	100%
almost certain	$95.0\% \pm 10.9\%$
highly likely	$90.0\% \pm 8.4\%$
very good chance	$80.0\% \pm 10.8\%$
likely	$70.0\% \pm 11.3\%$
probably	$70.0\% \pm 12.9\%$
probable	$70.0\% \pm 14.7\%$
better than even	$60.0\% \pm 9.1\%$
about even	$50.0\% \pm 4.9\%$
probably not	$25.0\% \pm 14.4\%$
unlikely	$20.0\% \pm 15.0\%$
little chance	$10.0\% \pm 12.2\%$
chances are slight	$10.0\% \pm 10.9\%$
improbable	$10.0\% \pm 17.5\%$
highly unlikely	$5.0\% \pm 17.3\%$
almost no chance	$2.0\% \pm 17.0\%$
impossible	0%

Table 5: The mapping of WEP to numeric probabilities by Fagen-Ulmschneider (2015) that we use to model uncertainty on the Bayesian networks in our dataset.

Hyp.-param.	ProbLog-FT	Regr.-FT	TP-FT
Learning rate	$5e-5$	$5e-5$	$5e-5$
LoRA rank	64	64	64
LoRA α	32	32	32
Batch size	4	4	4
Epochs	12	10	10
Num. GPUs	4	1	1

Table 6: Parameters for LoRA-based fine-tuning on QUITE.

D Exemplary Probability Calculation and Reasoning Steps

We now take a look at the smallest network instance in QUITE as shown in Figure 3 and manually execute the reasoning steps. The network is about the relationship of gallstones, flatulence, and amylase levels. We have three random variables

0: With a 15.31% likelihood, a patient is likely to have gallstones, while an 84.69% probability suggests the absence of gallstones in an individual.

1: If gallstones are present, there is a 39.25% chance that flatulence will also be present, and a 60.75% chance that flatulence will not be present.

2: If gallstones are absent, there is a 43.07% chance that flatulence will still be present, and a 56.93% chance that flatulence will not be present.

3: If gallstones are present, the likelihood of amylase levels between 0 and 299 is 93.46%, while elevated amylase levels of 300-499 units are slightly less likely at 4.67%, and extremely elevated amylase levels of 1400-500 units are even less likely at just 1.87%.

4: With no gallstones present, the probability of having amylase levels between 0 and 299 is significantly higher, at 97.30%, while elevated amylase levels of 300-499 units are less common at 1.69%, and extremely elevated amylase levels of 1400-500 units are even less likely at just 1.01%.

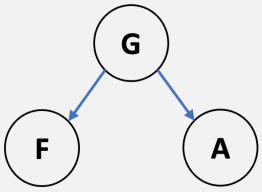


Figure 3: Exemplary network from QUITE about the relationship between gallstones, flatulence and amylase levels.

here which we are going to refer to as G (gallstones), F (flatulence), and A (amylase levels). It is important to mention that the models are only given the background premises, i.e., statements 0, 1, 2, 3, and 4. In a first step, they have to build up the graph shown, either as internal representations in their embeddings, as textual output in a chain-of-thought style of reasoning, or explicitly as ProbLog code. Furthermore, it is up to the model to determine not only the involved random variables, but also their possible states, i.e., Ω_i. For gallstones (G) and flatulence (F), it is a "yes" or "no" decision, i.e., $|\Omega_G| = |\Omega_F| = |\{yes, no\}| = 2$. Amylase levels show that QUITE also contains many categorical variables since A can take the values "0-299", "300-499", or "500-1400", i.e., $|\Omega_A| = |\{0-299, 300-499, 500-1400\}| = 3$.

Next, we look at a specific question-evidence (QE) pair with one observation (evidence) and one question:

- **Evidence**: *The patient has flatulence.*
- **Question**: *What is the likelihood of a patient having an amylase level between 1400 and 500 U/L?*

Before we can start calculating the answer to the question, we first need to understand the joint probability distribution that is represented by the network. It is a joint probability distribution over three random variables. Furthermore, the chain rule allows us to break down the joint probability into its single components:

$$\mathbb{P}(G, F, A) = \mathbb{P}(F|G) \cdot \mathbb{P}(A|G) \cdot \mathbb{P}(G)$$

We now calculate:

$$\mathbb{P}(A = 500-1400|F = yes)$$

According to the Bayes' theorem, this is equivalent to the joint probability over both variables divided by the condition:

$$\mathbb{P}(A = 500-1400|F = yes) = \frac{\mathbb{P}(A = 500-1400, F = yes)}{\mathbb{P}(F = yes)}$$

However, neither probability is explicitly given in the network. Instead, we only have conditional probabilities or the joint probability over all three random variables, not only two of them. To obtain the necessary probabilities, we need to "marginalize out" unwanted random variables by summing over all states:

$$\begin{aligned}&\mathbb{P}(A = 500-1400, F = yes)\\ &= \sum_G \mathbb{P}(A = 500-1400, F = yes, G)\\ &= \mathbb{P}(A = 500-1400, F = yes, G = yes)\\ &+\mathbb{P}(A = 500-1400, F = yes, G = no)\end{aligned}$$

We obtain $\mathbb{P}(A = 500-1400, F = yes, G = yes)$ by using the single conditional probabilities stated in the network:

$$\begin{aligned}&\mathbb{P}(A = 500-1400, F = yes, G = yes)\\ &= \mathbb{P}(F = yes|G = yes)\\ &\cdot\mathbb{P}(A = 500-1400|G = yes)\\ &\cdot\mathbb{P}(G = yes)\\ &= 0.3925 \cdot 0.0187 \cdot 0.1531 \approx 0.001124\end{aligned}$$

Doing the same for $G = no$ yields:

$$\mathbb{P}(A = 500 - 1400, F = yes, G = no)$$
$$= 0.4307 \cdot 0.0101 \cdot 0.8469 \approx 0.003684$$

$\mathbb{P}(A = 500 - 1400, F = yes)$ is now the sum of both parts, i.e.

$$\mathbb{P}(A = 500 - 1400, F = yes) \approx 0.004808$$

Obtaining $\mathbb{P}(F = yes)$ is as straightforward as just shown, but requires two marginalization steps, making it more calculation work:

$$\mathbb{P}(F = yes)$$
$$= \sum_{G,A} \mathbb{P}(F = yes, G, A)$$

Repeating the same steps from above, but with 6 summands since G can take 2 states and A 3 states yields the following probability:

$$\mathbb{P}(F = yes)$$
$$= \sum_{G,A} \mathbb{P}(F = yes, G, A)$$
$$= \mathbb{P}(F = yes, G = yes, A = 0 - 299)$$
$$+\mathbb{P}(F = yes, G = yes, A = 300 - 499)$$
$$+\mathbb{P}(F = yes, G = yes, A = 500 - 1400)$$
$$+\mathbb{P}(F = yes, G = no, A = 0 - 299)$$
$$+\mathbb{P}(F = yes, G = no, A = 300 - 499)$$
$$+\mathbb{P}(F = yes, G = no, A = 500 - 1400)$$
$$\approx 0.424856$$

Finally, we can obtain the answer to the question:

$$\mathbb{P}(A = 500 - 1400 | F = yes) = \frac{0.004808}{0.424856}$$
$$\approx \mathbf{0.01132}$$

Therefore, we conclude that the probability of having amylase levels between 500 and 1400 given the presence of flatulence is approximately 1.132%.

The models are now expected to either perform this calculation by themselves (either explicitly or implicitly) or parse the entire problem into ProbLog representation such that the ProbLog executable can calculate the actual answer.

```
0.1531::gallstones(patient).

0.3925::flatulence(patient) :- gallstones(patient).

0.4307::flatulence(patient) :- not gallstones(patient).

0.9346::amylase(patient, '0-299'); 0.0467::amylase(patient,
    '300-499'); 0.0187::amylase(patient, '500-1400') :-
    gallstones(patient).

0.9730::amylase(patient, '0-299'); 0.0169::amylase(patient,
    '300-499'); 0.0101::amylase(patient, '500-1400') :- not
     gallstones(patient).

evidence(flatulence(patient), true).

query(amylase(patient, '500-1400')).
```

Figure 4: Full ProbLog code for the gallstone-flatulence-amylase instance.

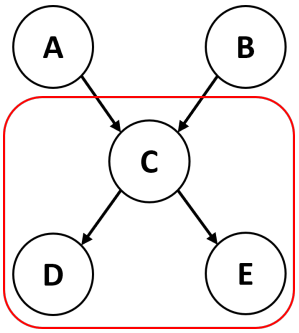


Figure 5: Example network for demonstrating our procedure of subsetting.

Figure 4 depicts how this mathematical problem is represented in ProbLog. The first five statements define the probabilistic model. We expect the model to perform semantic parsing from the natural language input to this structured representation. Each right-hand side represents the conditions for the left-hand side. Furthermore, since amylase levels is a categorical variable, it is necessary to connect all possible states via a semicolon in order to match them to the same probability distribution. This is an additional difficulty of QUITE since models can not rely on just writing down binary predicates. Next, the *evidence/2* predicate is used to set the observation. Here it is of key importance that the model only reuses predicates that were already defined in the parsed premises above. Finally, the question is defined using *query/1*. When calling ProbLog on this program, it outputs *amylase(patient,'500-1400'): 0.011316399*, which perfectly matches our calculation by hand.

E BN Subsetting

Assume we want to extract the network $D \leftarrow C \rightarrow E$ from the five-node network depicted in Figure 5.

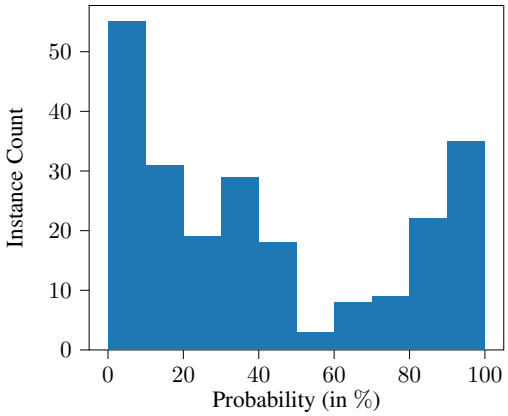


Figure 6: Distribution of probability values of the QE pairs in the test set of QUITE.

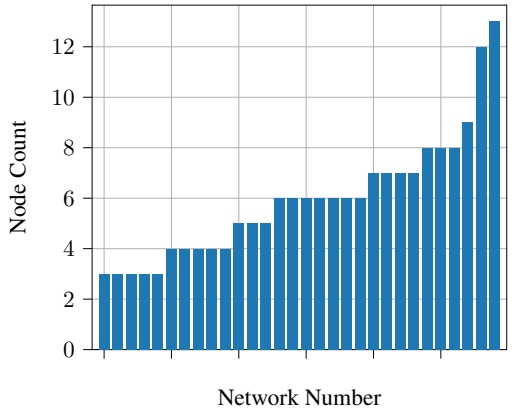


Figure 7: Node count for each of the 30 networks in QUITE.

It is not possible to just cut the connections $A \to C$ and $B \to C$ since node C only holds tables for CPDs that depend on A and B respectively. Therefore, we modify the subnetwork by marginalizing out A and B from the probability distribution of C:

$$\mathbb{P}(C) = \sum_{A,B} \mathbb{P}(C, A, B)$$
$$= \sum_{A,B} \mathbb{P}(C|A) \cdot \mathbb{P}(C|B) \cdot \mathbb{P}(A) \cdot \mathbb{P}(B)$$

F Further Dataset Statistics

In Appendix D, we used a three node network from QUITE. This was for demonstration purposes. However, QUITE contains networks of much larger sizes, i.e., with up to 13 nodes. Figure 7 shows the distribution of node counts across the 30 networks in the dataset. The largest network holds a joint probability distribution over 13 random variables. Also, more than half of the networks do have at least 5 nodes. The median size is 6, showing that the majority of our networks have of large size.

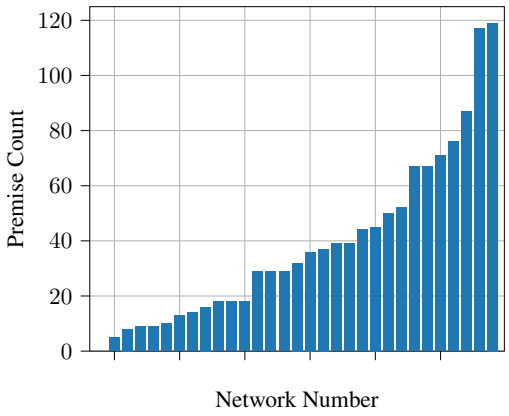


Figure 8: Premise count for each of the 30 networks in QUITE.

Figure 8 depicts the distribution of premise counts in QUITE. The median of the distribution is 34 and the maximum is 119. This shows that building up the probabilistic model from the set of premises is already a computationally demanding task and requires very long context understanding.

G Further Analysis

Figure 9 and Figure 10 sort the results of ProbLog-FT and CAUSALCOT on the ten networks in the test in ascending order by number of premises. There is a clear trend that shows that a growing number of background premises lead to an increasing amount of failure cases. This can be explained by the fact that having many background premises requires the model to work with an increasingly large message context of all already-parsed ProbLog premises.

When analyzing the performance for different numbers of states per (categorical) random variable (cf. Figure 11 and Figure 12), one cannot identify a clear trend. An increasing number of states seem more challenging, but we suspect that other factors such as complexity of the domain might play a bigger role. From this, we conclude that the amount of background premises seems to have a larger influence on the failure probability than the average amount of states per random variable.

H Data Generation Pipeline

Figure 13 displays an overview of our data generation pipeline. Every CPT entry is verbalized using the Mixtral LLM. Randomly sampled evidences and question nodes, which build the QE pairs in QUITE, are also transferred into natural language

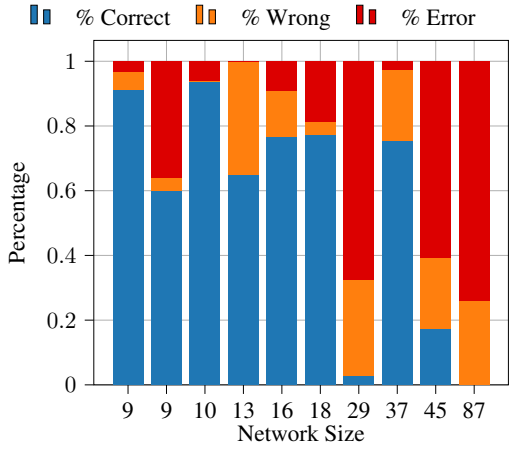


Figure 9: Results for Problog-FT for different network sizes.

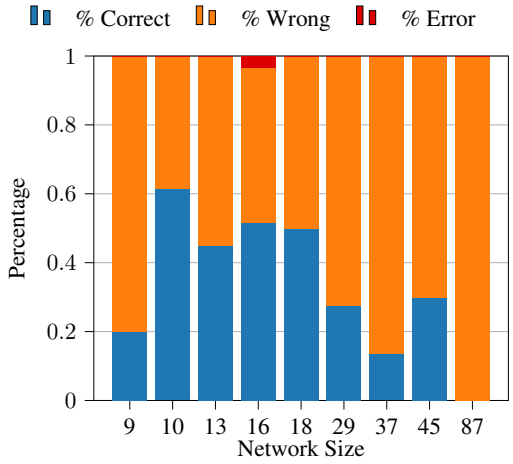


Figure 10: Results for CausualCoT with GPT4 for different network sizes.

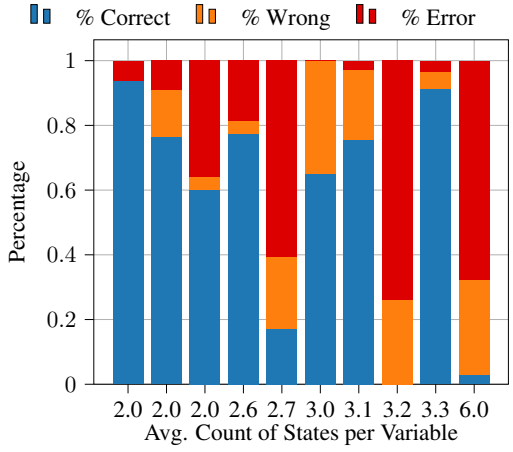


Figure 11: Results for Problog-FT for various numbers of states.

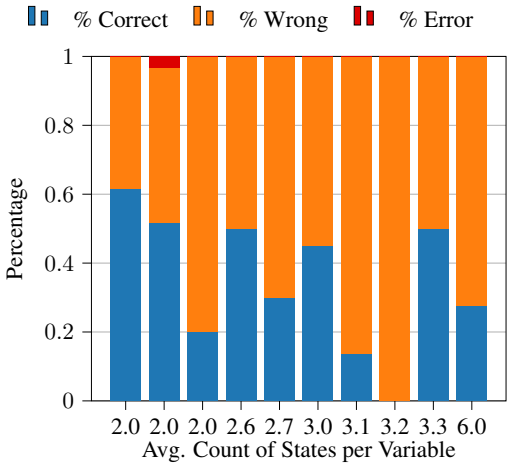


Figure 12: Results for CausualCoT with GPT4 for various numbers of states.

using Mixtral. For that, we prompt the LLM with the instruction to formulate grounded statements in the case of evidences and questions asking for probabilities in the case of queries. For each BN, we create an equivalent ProbLog representation that is used to generate the ground truth answer and to fine-tune the LLMs.

I CLADDER and BLInD

In this section, we provide one sample from CLADDER and BLInD each.

CLADDER uses pre-defined BN structures with three or four nodes. The following background premises are taken from a three-node network and represent the distribution over two binary random variables that can take the values *true* or *false*:

> *The overall probability of alarm set by husband is 3%. For husbands that don't set the alarm, the probability of ringing alarm is 74%. For husbands that set the alarm, the probability of ringing alarm is 22%.*

The corresponding question *Is ringing alarm more likely than silent alarm overall?* requires the following computation to obtain the answer: $\mathbb{P}(alarm = ringing) > \mathbb{P}(alarm = silent)$

One instance in BLInD is the following set of background premises:

> *If purple event is False, then grey event is True with probability of 39%. If purple event is False, then grey event is False with probability of 61%. If purple event*

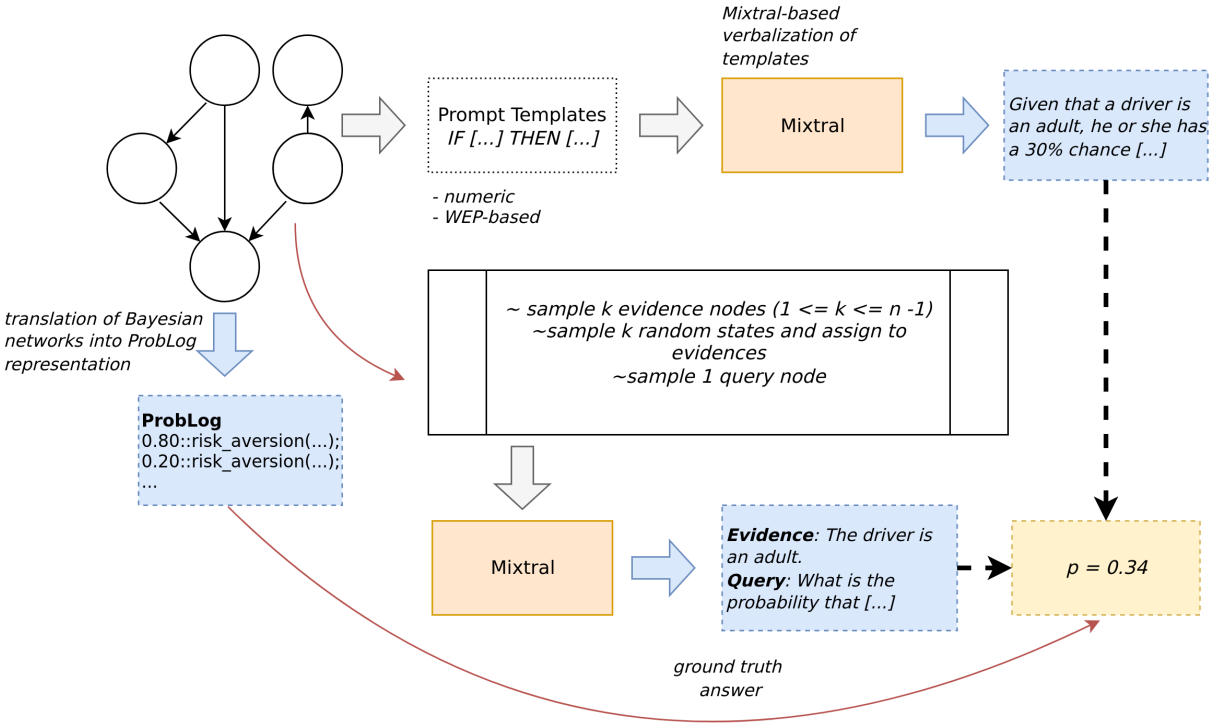


Figure 13: An schematic overview of our data generation pipeline.

is True, then grey event is True with probability of 3%. If purple event is True, then grey event is False with probability of 97%. purple event is true with probability of 55%. purple event is false with probability of 45%.

The corresponding question *What is the probability that grey event is True given that purple event is False?* requires the following computation to obtain the answer: $\mathbb{P}(grey = True|purple = False)$. Again, all random variables, in this case the color events, can only take the two states *true* or *false*.