Axiomatic Negation Coherence in Language Models: Evidence from FOLIO

Md Muntaqim Meherab[1*], **Naimur Rahman**[2]

[1]Department of Computer Science and Engineering, Daffodil International University
[2]Department of Software Engineering, Daffodil International University
meherab2305101354@diu.edu.bd, naimur15-9629@diu.edu.bd

Abstract

Large language models (LLMs) have quickly become the default tool for a wide range of NLP tasks, yet their logical behaviour is still poorly understood. Most existing evaluations focus on task accuracy on benchmarks, without asking whether a model's internal "beliefs" about statements are even coherent under basic logical principles. In this work, we take a small but concrete step in that direction. We view an LLM as a black-box function that maps a logical formula φ to a number $p(\varphi) \in [0, 1]$ that we interpret as the model's degree of belief that φ is true. Based on this view, we introduce a simple axiomatic framework that specifies how these degrees of belief should behave if they are to resemble a classical probability measure. We focus on one particularly transparent constraint: a negation-coherence axiom requiring $p(\varphi) + p(\neg\varphi) \leq 1$ for every formula φ. From this axiom we derive a per-instance violation score and an aggregate consistency metric. To make this concrete, we instantiate the framework on FOLIO, a first-order logic reasoning benchmark expressed in natural language. Using a small open-source chat model, TinyLlama-1.1B-Chat, we estimate $p(\varphi)$ and $p(\neg\varphi)$ from yes/no entailment judgments on a random subset of 200 FOLIO validation examples. The model turns out to be perfectly coherent with respect to our negation axiom: we observe zero violations in our sample. At the same time, its task performance is poor, with multi-valued accuracy of only 33%. In practice, the model almost always answers "no" to both the conclusion and its negation, thereby avoiding contradictions at the price of being largely uninformative. Our results highlight a simple but important point: logical coherence and reasoning competence are distinct properties. An LLM can be perfectly consistent with a basic logical axiom while still failing to make useful logical commitments. We argue that axiomatic consistency metrics such as ours offer a complementary lens on LLM behaviour, and we outline how the same framework can be extended to richer logical constraints and stronger models. We provide code and resources at: https://meherabb.github.io/Axiomatic/

*Corresponding author: Md Muntaqim Meherab. meherab2305101354@diu.edu.bd. This research was carried out at the DIU NLP & ML Research Lab, Daffodil International University.
Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Introduction

Over the last few years, large language models (LLMs) have moved from research curiosities to standard tools for natural language processing. Transformer architectures (Vaswani et al. 2017) and large-scale pre-training have enabled models such as GPT-3 to perform a wide range of tasks from instructions alone, often with little or no supervised fine-tuning (Brown et al. 2020). Prompting methods that explicitly elicit reasoning steps, such as chain-of-thought prompting (Wei et al. 2022), further show that these models can display strong performance on arithmetic, commonsense, and symbolic reasoning benchmarks.

Yet the logical behaviour of LLMs is still not well understood. Most evaluations focus on task-level metrics such as accuracy or F1, which tell us whether a model outputs the correct label but say little about whether its internal "beliefs" about different statements form a coherent whole. In practice it is easy to find cases where a model confidently endorses a statement in one context and rejects an equivalent statement in another, or assigns high plausibility to both a claim and its negation across different prompts. From the point of view of classical logic and probability theory, this clashes with even very basic consistency principles.

A growing line of work stresses LLMs on benchmarks that make logical structure explicit. LogiQA (Liu et al. 2020) and related reading comprehension datasets focus on deductive patterns embedded in short passages, while ProofWriter (Tafjord et al. 2021) explores whether neural models can generate implications and proofs over natural-language rule sets. FOLIO (Han et al. 2024) goes a step further by pairing natural-language premises and hypotheses with first-order logic (FOL) annotations that can be checked by an automated theorem prover. These resources map out where current models struggle with multi-step deduction, quantification, and the interaction of logical operators.

Most such evaluations still treat LLMs in the usual way: as functions that map an input problem to a single output label or string. In this paper we take a different view. We model an LLM as a black-box map from logical formulas φ to numbers $p(\varphi) \in [0, 1]$ that we interpret as degrees of belief. Concretely, we estimate $p(\varphi)$ from the model's yes/no answers to carefully phrased entailment questions in natural language, and then ask whether the resulting function p

obeys simple axioms that any classical probability measure over a Boolean algebra would satisfy.

We focus on perhaps the most elementary of these axioms: *negation coherence*. In classical probability, the probabilities of a statement and its negation must sum to one, and in particular can never sum to more than one. Translating this into our setting, we require that for every formula φ the model's degrees of belief satisfy

$$p(\varphi) + p(\neg\varphi) \leq 1.$$

This inequality is weaker than the exact equality enforced by Kolmogorov's axioms, but it already rules out a simple form of inconsistency: a model should not simultaneously treat both a claim and its negation as highly probable. From this axiom we derive a per-instance violation

$$v_{\text{neg}}(\varphi) = \max\big(0,\ p(\varphi) + p(\neg\varphi) - 1\big),$$

and define an aggregate negation-inconsistency score by averaging v_{neg} over a dataset.

To keep the empirical side manageable, we instantiate this framework on FOLIO (Han et al. 2024), which provides natural-language premises and conclusions together with formally verified FOL annotations. For each example, we query a small open-source chat model (TinyLlama-1.1B-Chat) twice: once about the conclusion φ and once about its negation $\neg\varphi$, using prompts that ask whether the statement is *logically entailed* by the premises. In this initial study, we adopt a simple approximation and treat deterministic yes/no answers as probabilities in $\{0, 1\}$.

The resulting picture is simple but revealing. On a random subset of 200 FOLIO validation instances, the model never asserts that both φ and $\neg\varphi$ are entailed, so the average negation-violation score is exactly zero. Under our axiom, the model is perfectly coherent. At the same time, its multi-valued accuracy (True / False / Uncertain) is only about 33%, and it almost never commits to saying that a conclusion is entailed or refuted when the ground-truth label is True or False. In practice, the model mostly answers "no" to both queries and therefore defaults to the label UNCERTAIN.

Taken together, these observations lead to the central message of the paper: logical coherence and reasoning competence are different notions. A model can satisfy a basic probabilistic consistency axiom in a trivial way by declining to make strong commitments. Our contribution is not to argue that TinyLlama is a good logician, but to show how a simple axiomatic lens makes this gap visible, and how the same framework can be extended to richer axioms (for example about entailment and conjunction) and applied to larger models and more diverse benchmarks.

Related Work

Our work sits at the intersection of three strands of research: (i) logical reasoning benchmarks for language models, (ii) hybrid neuro-symbolic approaches that couple LLMs with formal solvers, and (iii) efforts to measure or improve consistency and probabilistic behaviour in LLMs.

Logical Reasoning Benchmarks for LLMs

A number of benchmarks have been proposed to probe the logical reasoning abilities of neural models. LogiQA (Liu et al. 2020) collects multiple-choice questions from Chinese civil service examinations and focuses on deductive reading comprehension, highlighting that standard models still lag far behind human performance on carefully designed logical questions. ProofWriter (Tafjord et al. 2021) moves one step closer to formal reasoning, training generative models to derive implications and natural-language proofs from rule-based theories, and showing that transformers can emulate multi-step deduction over synthetic theories expressed in language.

More recently, FOLIO (Han et al. 2024) offers a bridge between natural language and first-order logic: each problem consists of premises and hypotheses in plain text, together with a formal FOL representation that can be checked by a theorem prover. This dual view makes it possible to evaluate models both as natural language reasoners and against a formal ground truth. LogicBench (Parmar et al. 2024) takes a complementary approach, constructing a suite of natural language tasks that each isolate a single inference rule. The authors show that even state-of-the-art models such as GPT-4 and Llama-2 struggle when reasoning requires careful handling of negation or non-monotonic patterns.

Our experimental setup is deliberately modest compared to these large-scale benchmarks: we use only a small subset of FOLIO and a single open-source model. The difference lies in what we measure. Rather than proposing yet another benchmark, we treat FOLIO as a source of logically grounded examples on which to instantiate an axiomatic consistency metric. In that sense, our work is closer in spirit to diagnostic tools than to leaderboard-style evaluation.

Hybrid Neuro-Symbolic Reasoning

A second line of work seeks to improve logical reasoning by combining LLMs with symbolic tools. Logic-LM (Pan et al. 2023) is a prominent example: it uses an LLM to translate natural language problems into a logical form and then delegates the actual deduction to a symbolic solver, with a refinement loop that uses solver feedback to correct the logical formalisation. Logic-LM achieves large gains over pure prompting-based baselines across several logical reasoning datasets, including ProofWriter and FOLIO. Similar ideas appear in more recent systems that pair LLMs with first-order provers or SMT solvers for theorem proving and program verification, but the core pattern is the same: the neural model proposes, the symbolic engine disposes.

These hybrid approaches address a different question from ours. They aim to *improve* the reasoning performance of LLM-centric systems by offloading hard logic to specialized tools. We, by contrast, do not try to fix the model. We keep the model as-is and ask a more basic question: if we interpret its yes/no answers as degrees of belief, do those degrees satisfy even very simple logical constraints, such as a negation inequality? Our framework could, in principle, be applied to the outputs of a neuro-symbolic pipeline as well, but that is beyond the scope of this initial study.

Consistency, Calibration, and Probabilistic Behaviour

There is also a growing interest in understanding how consistent and well-calibrated LLMs are. Lyu et al. (Lyu et al. 2025) propose to calibrate LLMs using what they call *sample consistency*: instead of relying on hidden probabilities, they estimate confidence from the agreement pattern across multiple sampled generations. Their work is squarely focused on confidence estimation and calibration error, but it already suggests that consistency constraints can be a useful signal when we only have black-box access to the model.

At the same time, researchers have begun to evaluate how language models handle explicitly probabilistic tasks. Paruchuri et al. (Paruchuri et al. 2024) study how models estimate probabilities, percentiles, and samples from idealised and real-world distributions, finding that models can display non-trivial probabilistic reasoning when given appropriate context, but also systematic biases. These works treat probability as the object of reasoning: the model is asked *about* probabilities. Our work is closer to probabilistic semantics: we use a probability-like quantity $p(\varphi)$ as a way to summarise what the model "believes" about the truth of φ.

Most closely related to our aims are recent efforts to enforce or analyse logical consistency directly. Some approaches integrate logical constraints into training objectives using semantic losses or neuro-symbolic architectures to encourage models to satisfy integrity constraints over facts and rules, thereby improving logical agreement with a knowledge base. Others propose metrics for logical consistency in tasks such as preference ranking, document ordering, or temporal reasoning, and show that higher consistency often correlates with robustness. Our contribution is more minimalistic: we restrict attention to a single inequality derived from classical probability theory and show how even this one axiom can be turned into a concrete, per-instance violation score over a benchmark like FOLIO.

Summary

Table 1 summarises where our work fits relative to some of the most relevant prior lines. Existing benchmarks mainly ask whether models produce the right label. Hybrid systems ask whether we can *fix* models by coupling them to symbolic tools. Calibration and probabilistic reasoning work ask whether we can trust the model's confidence. We instead treat the model's degrees of belief as a mathematical object and inspect whether they obey a basic negation axiom.

Axiomatic Framework

In this section we spell out the simple mathematical lens we use to inspect the logical behaviour of a language model. The key idea is to treat the model as if it equips each formula φ with a "degree of belief" $p(\varphi) \in [0, 1]$, and then to ask whether this degree of belief obeys basic constraints that any classical probability measure would satisfy. We first describe how we obtain $p(\varphi)$ from an LLM, then introduce a small family of logical axioms over p, and finally define violation scores that can be computed on a benchmark such as FOLIO.

LLM as a Probabilistic Truth Function

Let $\mathcal{L}$ be a propositional or quantifier-free first-order language generated from a set of atomic propositions using the standard connectives $\neg$ (negation), $\wedge$ (conjunction), $\vee$ (disjunction), and $\rightarrow$ (implication). We think of formulas in $\mathcal{L}$ as abstractions of natural language statements that appear in datasets such as FOLIO.

We model a language model as a black-box function

$$f_\theta : \mathcal{L} \rightarrow [0, 1],$$

where θ denotes the model parameters and $f_\theta(\varphi)$ is interpreted as the model's degree of belief that φ is true, given a fixed context. In an ideal world, one might hope that f_θ behaves like a probability measure over a Boolean algebra of events induced by $\mathcal{L}$, but in practice we will only enforce a small number of necessary conditions.

Concretely, we do not have direct access to $f_\theta(\varphi)$; instead we estimate a surrogate quantity $p(\varphi)$ from the model's answers to yes/no questions. For a given natural language problem, we consider a set of premises Prem and a conclusion c, and construct a prompt of the form:

> You are given several premises and one conclusion.
> Assume that *all* premises are true.
>
> Premises: ...
>
> Conclusion: ...
>
> Is the conclusion logically entailed by the premises?
> Answer YES or NO.

The model's answer induces a Bernoulli random variable $Y(\varphi)$, where φ encodes the formal statement "c is logically entailed by Prem". In principle, we could query the model multiple times with controlled sampling and set $p(\varphi) = \Pr(Y(\varphi) = 1)$, or use token log-probabilities for the words "YES" and "NO". In this initial study we adopt the simplest possible approximation and take

$$p(\varphi) = \begin{cases} 1 & \text{if the model outputs YES,} \\ 0 & \text{otherwise.} \end{cases}$$

We repeat the same process for the negated conclusion $\neg\varphi$, by asking whether the *negation* of the conclusion is logically entailed by the premises. This gives us a pair $(p(\varphi), p(\neg\varphi)) \in \{0, 1\}^2$ for each example. While this binary approximation is clearly coarse, it is enough to make the axioms and violation scores that follow concrete and easy to compute.

Logical Axioms for Degrees of Belief

We now describe the axioms that we would like our degree-of-belief function p to satisfy. Throughout this subsection, we imagine an idealised setting where p is the restriction of a classical probability measure P over a Boolean algebra of events. In such a setting, the following statements all hold, and they serve as sanity checks for the behaviour of an LLM viewed through p.

Work	Main focus	Setting	Relation to ours
LogiQA (Liu et al. 2020)	Logical reading comprehension	MCQ QA over texts	Benchmarking deductive reading; we use FOLIO instead.
ProofWriter (Tafjord et al. 2021)	Generating implications and proofs	Synthetic rules in NL	Shows transformers can emulate deduction; we focus on axiomatic consistency.
FOLIO (Han et al. 2024)	FOL-annotated reasoning	NL + FOL proofs	Provides the dataset and formal ground truth we build on.
LogicBench (Parmar et al. 2024)	Systematic logical reasoning eval	25 inference patterns	Evaluates accuracy across rules; we measure consistency over one axiom.
Logic-LM (Pan et al. 2023)	LLM + symbolic solver	Neuro-symbolic pipeline	Improves reasoning via external tools; we analyse raw model beliefs.
Lyu et al. (Lyu et al. 2025)	Consistency-based calibration	Multiple LLMs, 9 datasets	Uses sample consistency for calibration; we use axioms over $p(\varphi)$.

Table 1: Selected related work and how it compares to our focus on axiomatic logical consistency over FOLIO.

Axiom A1 (Normalization). For every formula $\varphi \in \mathcal{L}$,

$$0 \leq p(\varphi) \leq 1.$$

This axiom is almost tautological in our setup, since we construct $p(\varphi)$ to lie in $[0, 1]$ by design. We include it mainly to fix notation and emphasize that we are thinking in probabilistic terms.

Axiom A2 (Negation coherence). For every formula $\varphi \in \mathcal{L}$,

$$p(\varphi) + p(\neg\varphi) \leq 1.$$

Intuitively, this axiom says that the model should not simultaneously assign high belief to a statement and to its negation. In classical probability, if p is induced by a measure P, then $P(\varphi)$ and $P(\neg\varphi)$ sum to exactly one because the corresponding events are complements. We relax this to an inequality to allow some slack, especially given that our estimates of p are noisy and obtained from a discrete prompting protocol rather than from an explicit probability space.

Axiom A3 (Monotonicity under entailment). Let $\varphi, \psi \in \mathcal{L}$. If φ logically entails ψ (denoted $\varphi \models \psi$), then

$$p(\varphi) \leq p(\psi).$$

The intuition is straightforward: if every world that satisfies φ also satisfies ψ, then ψ should be at least as plausible as φ. In the ideal probabilistic setting, this follows from the fact that the event corresponding to φ is a subset of the event corresponding to ψ, and therefore has no greater probability mass.

Axiom A4 (Conjunction lower bound). For any pair of formulas $\varphi, \psi \in \mathcal{L}$,

$$p(\varphi \wedge \psi) \geq p(\varphi) + p(\psi) - 1.$$

This inequality is a direct analogue of the classical Bonferroni inequality for two events. It reflects the idea that the degree of belief in the conjunction cannot be arbitrarily small relative to the degrees of belief in the individual components. Once again, in a full probabilistic model this inequality is guaranteed to hold; violations in our setting indicate that the model's degrees of belief are not in line with any underlying probability measure.

In this paper we *only* measure violations of Axiom A2 in our experiments, primarily because it can be instantiated on FOLIO without additional machinery. Axioms A3 and A4 are included to make the framework more complete and to sketch the kinds of constraints that could be explored in future work.

Violation Functions and Aggregate Scores

To turn the axioms above into quantities that can be computed on a dataset, we define a non-negative violation function for each axiom. For the negation coherence axiom, the per-instance violation is

$$v_{\text{neg}}(\varphi) = \max\big(0,\ p(\varphi) + p(\neg\varphi) - 1\big).$$

By construction, $v_{\text{neg}}(\varphi) = 0$ whenever Axiom A2 holds, and becomes positive exactly when $p(\varphi) + p(\neg\varphi) > 1$. In our binary approximation where $p(\varphi), p(\neg\varphi) \in \{0, 1\}$, this simplifies further: $v_{\text{neg}}(\varphi)$ is equal to 1 if the model answers "yes" to both the conclusion and its negation, and 0 otherwise.

For completeness, we also define hypothetical violation functions for the other two axioms:

$$v_{\text{ent}}(\varphi, \psi) = \max\big(0,\ p(\varphi) - p(\psi)\big),$$
$$v_{\wedge}(\varphi, \psi) = \max\big(0,\ p(\varphi) + p(\psi) - 1 - p(\varphi \wedge \psi)\big).$$

Here v_{ent} measures how badly the monotonicity requirement is violated for a known entailment pair $\varphi \models \psi$, and $v_{\wedge}$ measures how much the conjunction lower bound is violated for a given pair (φ, ψ).

Given a dataset of formulas (and, where needed, entailment relations between them), we can form aggregate scores by averaging these violation functions. For the negation axiom, the aggregate score is

$$V_{\text{neg}} = E\big[v_{\text{neg}}(\varphi)\big],$$

where the expectation is taken with respect to a distribution over formulas in the dataset. In practice, V_{neg} is just the empirical mean of $v_{\text{neg}}(\varphi)$ across examples. Analogous definitions would apply for V_{ent} and $V_{\wedge}$ if we had access to the relevant triples.

It is immediate from the definitions that $V_{\text{neg}} = 0$ if and only if all examined instances satisfy Axiom A2 exactly, and

similarly for the other axioms. In that sense, V_{neg} acts as a soft measure of how far a model's degree-of-belief function is from obeying the negation constraint on a given dataset. In our experiments, we focus exclusively on v_{neg} and V_{neg}, since they already produce an informative picture on FOLIO.

Summary of the Framework

Table 2 summarises the key elements of the framework introduced above. The rest of the paper shows how these pieces can be instantiated on FOLIO with a small open-source model.

Experimental Setup

Our empirical goal is modest: we want to see what our negation-coherence axiom looks like in practice on a real logical reasoning benchmark, using an off-the-shelf open model. This section describes the dataset, the model, the prompting protocol we use to estimate $p(\varphi)$ and $p(\neg\varphi)$, and the derived evaluation metrics.

Dataset: FOLIO

We base our study on the FOLIO benchmark (Han et al. 2024). Each FOLIO instance consists of:

- a set of natural-language premises,
- a natural-language hypothesis (conclusion),
- a label indicating whether the hypothesis is TRUE, FALSE, or UNCERTAIN with respect to the premises, and
- a parallel annotation in first-order logic (FOL) together with a proof or refutation generated by an automated theorem prover.

The FOL layer gives us a formally verified notion of entailment and refutation, which fits our axiomatic perspective very well.

In this work we focus on the validation split released by the authors. From this split we draw a uniform random sample of $N = 200$ instances, without replacement. We use the natural-language premises and hypotheses to query the model, and the FOL-backed labels as our ground truth about whether the conclusion is entailed (TRUE), refuted (FALSE), or neither (UNCERTAIN).

Model

For the model, we deliberately choose a small open-source chat model rather than a large proprietary system. Our aim is not to chase benchmark state-of-the-art, but to illustrate how the axiomatic framework behaves on a realistic but lightweight setup that can be reproduced on a single GPU.

Concretely, we use *TinyLlama-1.1B-Chat*, a roughly 1.1B-parameter Llama-style model available on public model hubs. The model is instruction-tuned and comes with a simple chat template based on special tokens such as `<|im_start|>` and `<|im_end|>`. All experiments are run in a Google Colab environment with a single GPU; no fine-tuning or additional training is performed.

We stress that TinyLlama is not advertised as a logical reasoner. If anything, it is a deliberately weak baseline. This choice is intentional: if our framework can already uncover interesting structure at this scale, it becomes even more compelling as a diagnostic tool for larger models.

Prompting Protocol and Estimation of $p(\varphi)$

For each of the 200 sampled FOLIO instances, we construct a short "story" that lists the premises and the conclusion in a fixed template. We then ask two questions per instance:

Entailment query (φ). We ask whether the conclusion is logically entailed by the premises. The prompt follows the pattern

> You are given several premises and one conclusion. Assume that *all* premises are true.
>
> Premises: 1. …
> 2. …
> ⋮
>
> Conclusion: …
>
> Question: Is the conclusion logically entailed by the premises? In other words, must the conclusion be true in every model where all premises are true? Answer YES or NO.

This question corresponds to a formula φ that we can think of as "the conclusion is entailed by the premises".

Negation query ($\neg\varphi$). We ask the same question, but about the negation of the conclusion:

> Question: Is the *negation* of the conclusion logically entailed by the premises? In other words, must the conclusion be false in every model where all premises are true? Answer YES or NO.

This corresponds to the formal statement $\neg\varphi$.

The model is instructed, via a system-style prefix, to respond with a single word, either `YES` or `NO`. In practice, the decoded output can contain extra whitespace or punctuation, so we normalise by converting the first non-whitespace token to uppercase and checking whether it begins with `YES` or `NO`.

Given these two answers, we obtain a binary approximation of the degree-of-belief function:

$$p(\varphi) = \begin{cases} 1 & \text{if the model answers YES} \\ & \text{to the entailment query,} \\ 0 & \text{otherwise,} \end{cases}$$
$$p(\neg\varphi) = \begin{cases} 1 & \text{if the model answers YES} \\ & \text{to the negation query,} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

This approximation is intentionally crude. It ignores token-level probabilities and treats only the final discrete decision as signal. For the purposes of this first study, the advantage is simplicity: $p(\varphi)$ and $p(\neg\varphi)$ are always in $\{0, 1\}$, which makes the violation score easy to interpret.

Component	Informal description	Role in this paper
$p(\varphi)$	Degree of belief that formula φ is true, estimated from yes/no answers.	Basic object we measure and constrain.
Axiom A2	Negation coherence: $p(\varphi) + p(\neg\varphi) \leq 1$.	Only axiom we instantiate empirically.
$v_{\text{neg}}(\varphi)$	Per-instance violation $\max(0, p(\varphi) + p(\neg\varphi) - 1)$.	Flags cases where model endorses both a claim and its negation.
V_{neg}	Average of $v_{\text{neg}}(\varphi)$ over a dataset.	Our main aggregate consistency score.

Table 2: Summary of the main elements of our axiomatic framework. In experiments we focus on the negation axiom A2 and its violation scores v_{neg} and V_{neg}.

Derived Labels and Evaluation Metrics

The FOLIO ground truth tells us, for each instance, whether the natural-language conclusion is logically entailed (TRUE), refuted (FALSE), or neither (UNCERTAIN). From this we derive two Boolean flags:

$$\text{gold\_}\varphi_{\text{true}} = \begin{cases} \text{True} & \text{if label = TRUE,} \\ \text{False} & \text{otherwise,} \end{cases}$$
$$\text{gold\_}\neg\varphi_{\text{true}} = \begin{cases} \text{True} & \text{if label = FALSE,} \\ \text{False} & \text{otherwise.} \end{cases} \tag{2}$$

These indicate whether the conclusion or its negation is in fact entailed by the premises according to the FOL proof.

On top of this structure we define two kinds of metrics.

Axiomatic consistency metrics. For each instance we compute the negation-coherence violation

$$v_{\text{neg}}(\varphi) = \max\big(0,\ p(\varphi) + p(\neg\varphi) - 1\big).$$

In our binary setting, $v_{\text{neg}}(\varphi)$ is 1 exactly when the model answers YES to both queries, and 0 otherwise. The aggregate negation-inconsistency score V_{neg} is the empirical mean of $v_{\text{neg}}(\varphi)$ across the 200 examples. We also report the fraction of instances where $(p(\varphi), p(\neg\varphi)) = (1, 1)$, which we refer to as the *inconsistent* fraction.

Task-oriented performance metrics. Although our main focus is axiomatic consistency, it is helpful to relate this to more familiar task metrics. We define:

- φ **accuracy on TRUE:** the fraction of instances with gold label TRUE for which $p(\varphi) = 1$.
- φ **over-claim rate on FALSE:** the fraction of instances with gold label FALSE for which $p(\varphi) = 1$, i.e., how often the model incorrectly asserts entailment when the conclusion is actually refuted.
- $\neg\varphi$ **accuracy on FALSE:** the fraction of FALSE instances for which $p(\neg\varphi) = 1$.

Finally, we combine the two answers into a four-way predicted label:

$$\text{pred} = \begin{cases} \text{TRUE} & \text{if } p(\varphi) = 1,\ p(\neg\varphi) = 0, \\ \text{FALSE} & \text{if } p(\varphi) = 0,\ p(\neg\varphi) = 1, \\ \text{UNCERTAIN} & \text{if } p(\varphi) = 0,\ p(\neg\varphi) = 0, \\ \text{INCONSISTENT} & \text{if } p(\varphi) = 1,\ p(\neg\varphi) = 1, \end{cases}$$

and compute a multi-valued accuracy by comparing `pred` to the ground-truth label in {TRUE, FALSE, UNCERTAIN}. This accuracy does not directly reflect the negation axiom, but it helps to put the consistency scores into perspective: a model can appear perfectly coherent while still performing poorly on the underlying task.

Summary

Table 3 summarises the main elements of our experimental setup, linking each configuration choice to its role in the analysis.

Results and Discussion

We now report what our axiomatic lens reveals about TinyLlama-1.1B-Chat on the sampled FOLIO instances. We first summarise the quantitative results, then look more closely at the patterns behind the numbers, and finally discuss what this small case study suggests about the relationship between logical coherence and reasoning competence.

Main Quantitative Findings

Table 4 summarises the core metrics introduced in the *Axiomatic Framework* section and the *Experimental Setup* section. All numbers are computed on the random subset of $N = 200$ FOLIO validation examples.

The picture is quite stark. On the one hand, the model is perfectly coherent with respect to the negation axiom A2 on this subset: the average violation rate is exactly zero, and we never observe a case where the model answers YES to both φ and $\neg\varphi$ for the same premises. On the other hand, the task-oriented metrics are poor. When the ground-truth label is TRUE, the model only answers YES to the entailment query in 1.4% of cases. When the label is FALSE, it almost never asserts either entailment or refutation: it over-claims entailment in only 1.7% of such cases, and it never answers YES to the negation query when refutation is correct. The resulting multi-valued accuracy is 33%, which is roughly on the order of the marginal frequency of the UNCERTAIN label in our sample.

How the Model Uses φ and $\neg\varphi$

To understand these numbers, it is helpful to look at how often different combinations of answers occur. Figure 1 compares the distribution of gold labels with the distribution of model predictions derived from $(p(\varphi), p(\neg\varphi))$. The model

Component	Configuration	Purpose
Dataset	FOLIO validation split; random subset of $N = 200$ examples	Provides natural-language premises/-conclusions and FOL-backed labels (TRUE/FALSE/UNCERTAIN).
Model	TinyLlama-1.1B-Chat; no fine-tuning	Lightweight open model to illustrate the axiomatic framework.
Queries	Two prompts per instance: entailment of c (query for φ) and entailment of $\neg c$ (query for $\neg\varphi$)	Used to estimate $p(\varphi), p(\neg\varphi) \in \{0, 1\}$.
Consistency metric	$v_{\text{neg}}(\varphi)$ and V_{neg}	Measure violations of the negation-coherence axiom.
Task metrics	φ accuracy on TRUE; φ over-claim on FALSE; $\neg\varphi$ accuracy on FALSE; multi-valued accuracy	Relate axiomatic consistency to standard task performance.

Table 3: Summary of experimental setup on FOLIO.

Metric	Value
# examples	200
Negation violation rate $E[\max(0, p(\varphi) + p(\neg\varphi) - 1)]$	0.000
Fraction INCONSISTENT (YES to both φ and $\neg\varphi$)	0.000
φ accuracy on gold label = TRUE	0.014
φ over-claim rate on gold label = FALSE	0.017
$\neg\varphi$ accuracy on gold label = FALSE	0.000
Multi-valued accuracy (TRUE/FALSE/UNCERTAIN)	0.330

Table 4: Main results for TinyLlama-1.1B-Chat on a random subset of 200 FOLIO validation instances. The model is queried twice per example, once about the conclusion φ and once about its negation $\neg\varphi$.

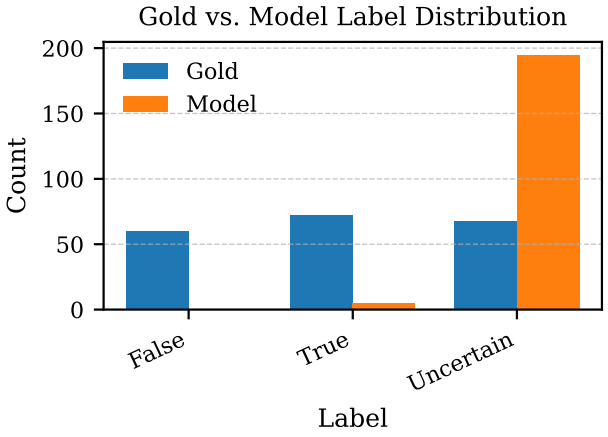

Figure 1: Distribution of ground-truth labels versus TinyLlama's predictions on the 200-example FOLIO subset. The model predicts UNCERTAIN for most inputs, regardless of the gold label.

strongly favours the UNCERTAIN label and rarely predicts TRUE or FALSE, even when the FOL prover indicates that the conclusion is entailed or refuted. This explains why the φ accuracy on TRUE and the $\neg\varphi$ accuracy on FALSE are so low: the model is simply not willing to commit to strong logical judgments.

Figure 2 zooms in on the joint pattern of answers to the two queries. Almost all mass lies in the (NO, NO) cell: the model typically answers NO to both "is the conclusion entailed?" and "is the negation of the conclusion entailed?". The other three patterns—(YES, NO), (NO, YES), and (YES, YES)—are rare, with (YES, YES) not appearing at all in our sample. In terms of the degrees-of-belief function p, this means that for most instances we have $p(\varphi) = p(\neg\varphi) = 0$.

From the perspective of our negation axiom, this behaviour is perfectly safe. Since $p(\varphi) + p(\neg\varphi) = 0$ almost always, the negation violation $v_{\text{neg}}(\varphi)$ is identically zero.

Figure 3 shows the distribution of v_{neg} as a histogram: every example lands at exactly 0. Under our metric, TinyLlama looks like an exemplary citizen with respect to negation coherence.

Coherence Without Competence

The problem, of course, is that this coherence is largely vacuous. A model that always answered NO to everything would also achieve $V_{\text{neg}} = 0$ by construction, but it would not be very useful as a reasoner. Our results suggest that TinyLlama is not far from this degenerate behaviour on FOLIO. It avoids logical contradictions by declining to assert either side of a potential opposition. In other words, it behaves like an extremely cautious agent that rarely sticks its neck out.

This tension between coherence and competence is the central takeaway of our small study. If we only looked at the negation violation scores, we might conclude that the model's internal degrees of belief are nicely aligned with a simple probabilistic axiom. Once we place these scores next to the task metrics, a different picture emerges: the model appears consistent mostly because it refuses to commit.

Seen in this light, our framework serves as a reminder that axioms must be interpreted in context. Negation coherence is a necessary but far from sufficient property for a reasonable belief state. A fully informative reasoner would often need to assign high probability to φ when the FOL ground truth says that the conclusion is entailed, and high probability to $\neg\varphi$ when the conclusion is refuted. On such instances,

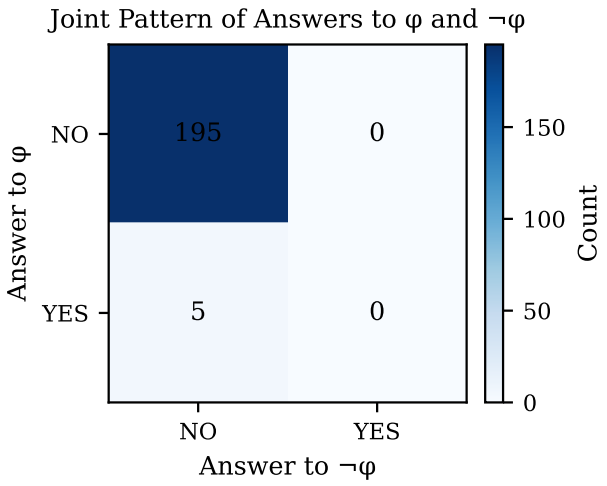

Figure 2: Joint pattern of answers to the entailment query φ and the negation query $\neg\varphi$. Almost all mass lies in the (NO, NO) cell, indicating that the model typically declines to assert either entailment or refutation.

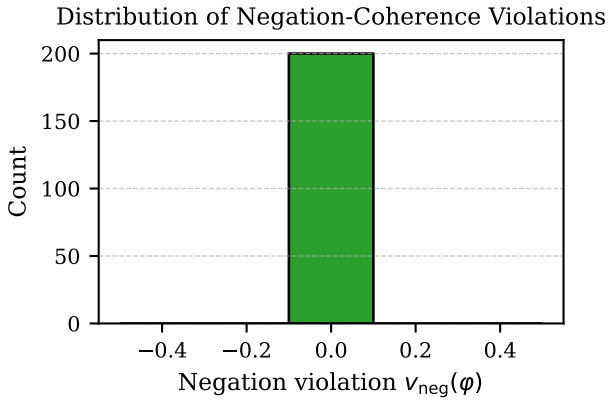

Figure 3: Histogram of negation-coherence violations $v_{\text{neg}}(\varphi)$ across the 200 examples. All mass is at 0, confirming that the model never simultaneously asserts both φ and $\neg\varphi$.

we would expect $p(\varphi)$ and $p(\neg\varphi)$ to be far from zero, and yet still satisfy $p(\varphi) + p(\neg\varphi) \leq 1$. Our TinyLlama case study lives in the opposite regime: the axiom is satisfied in a trivial way, not through fine-grained balancing of belief, but through near-total abstention.

Relation to Calibration and Consistency Work

These observations resonate with recent work on calibration and consistency for LLMs. Lyu et al. (Lyu et al. 2025) show that aggregating over multiple sampled answers can yield better-calibrated confidence estimates than raw token probabilities; their notion of *sample consistency* is linked to how stable a model's decisions are under small perturbations. Our experiment highlights a different aspect: even when decisions are stable, the underlying belief function can satisfy certain formal constraints for trivial reasons. Likewise, studies of probabilistic reasoning (Paruchuri et al. 2024) show that LLMs can sometimes reason sensibly about probabilities when those are explicit in the input. In our case, probability appears only implicitly through $p(\varphi)$, and the model takes the easy way out.

We do not claim that these negative results are representative of all models or tasks. A larger model, or one trained with explicit logical supervision, might well achieve both low violation scores and high task accuracy. What our study does show is that axiomatic metrics like V_{neg} can add a valuable dimension to the evaluation toolbox. They can tell us *how* a model stays out of trouble: by balancing its beliefs in a probabilistically sensible way, or by mostly sitting on the fence.

Limitations and Opportunities

Our experiments are intentionally narrow: one model, one benchmark, one axiom, and a binary estimate of $p(\varphi)$ from yes/no answers. All of these choices can be relaxed. A natural next step is to use token-level probabilities or multiple samples to obtain smoother $p(\varphi)$ and $p(\neg\varphi)$, and to test whether larger models still satisfy the negation axiom when they make stronger commitments. Another direction is to move beyond negation and study axioms about entailment or conjunction, which would require constructing groups of related formulas from FOLIO or similar datasets.

Even with these caveats, the main point remains: logical consistency under simple axioms is not the same as strong logical reasoning. A small, open model like TinyLlama can look perfectly coherent under one axiom while contributing little on a demanding reasoning benchmark. Understanding this gap, rather than only reporting accuracy, is important for building more trustworthy and interpretable language models.

Conclusion and Future Work

We took a small step towards understanding the logical behaviour of large language models by treating them as assigning a degree of belief $p(\varphi)$ to each statement φ and asking whether these beliefs obey a simple negation axiom. This yielded a per-instance violation score $v_{\text{neg}}(\varphi)$ and an aggregate metric V_{neg}, which we computed on FOLIO using yes/no entailment queries.

On 200 FOLIO examples with TinyLlama-1.1B-Chat, the model never asserted both a conclusion and its negation, so the negation axiom was perfectly satisfied, yet it rarely endorsed entailment or refutation when those were actually correct, leading to low task accuracy and a strong bias towards the UNCERTAIN label. This gap between "does not contradict itself" and "often reasons correctly" is the central message of our study.

Looking ahead, richer axioms (for example about entailment or conjunction), smoother estimates of $p(\varphi)$, and experiments with larger models or hybrid neuro-symbolic systems could show when low violation scores reflect genuinely

well-structured beliefs and when they simply reflect cautious behaviour. One promising direction is to treat the axioms as defining a convex region of consistent belief states and project a model's raw belief vector into that region, as a form of post-hoc repair and as a tool for analysing how far current systems are from basic logical ideals.

References

[Vaswani et al. 2017] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention Is All You Need. *Advances in Neural Information Processing Systems*, 30.

[Brown et al. 2020] Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; et al. 2020. Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, 33.

[Wei et al. 2022] Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E. H.; Le, Q. V.; and Zhou, D. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *Advances in Neural Information Processing Systems*.

[Han et al. 2024] Han, S.; Schoelkopf, H.; Zhao, Y.; Qi, Z.; Riddell, M.; Zhou, W.; Coady, J.; Peng, D.; Qiao, Y.; Benson, L.; Sun, L.; Wardle-Solano, A.; Szabó, H.; Zubova, E.; Burtell, M.; Fan, J.; Liu, Y.; Wong, B.; Sailor, M.; Ni, A.; Nan, L.; Kasai, J.; Yu, T.; Zhang, R.; Fabbri, A.; Kryscinski, W. M.; Yavuz, S.; Liu, Y.; Lin, X. V.; Joty, S.; Zhou, Y.; Xiong, C.; Ying, R.; Cohan, A.; and Radev, D. 2024. FOLIO: Natural Language Reasoning with First-Order Logic. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Miami, Florida, USA: Association for Computational Linguistics.

[Liu et al. 2020] Liu, J.; Cui, L.; Liu, H.; Huang, D.; Wang, Y.; and Zhang, Y. 2020. LogiQA: A Challenge Dataset for Machine Reading Comprehension with Logical Reasoning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*.

[Tafjord et al. 2021] Tafjord, Ø.; Dalvi, B.; and Clark, P. 2021. ProofWriter: Generating Implications, Proofs, and Abductive Statements over Natural Language. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*.

[Parmar et al. 2024] Parmar, M.; Patel, N.; Varshney, N.; Nakamura, M.; Luo, M.; Mashetty, S.; Mitra, A.; and Baral, C. 2024. LogicBench: Towards Systematic Evaluation of Logical Reasoning Ability of Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 13679–13707. Bangkok, Thailand: Association for Computational Linguistics.

[Pan et al. 2023] Pan, L.; Albalak, A.; Wang, X.; and Wang, W. Y. 2023. Logic-LM: Empowering Large Language Models with Symbolic Solvers for Faithful Logical Reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 3806–3824. Singapore: Association for Computational Linguistics.

[Lyu et al. 2025] Lyu, Q.; Shridhar, K.; Malaviya, C.; Zhang, L.; Elazar, Y.; Tandon, N.; Apidianaki, M.; Sachan, M.; and Callison-Burch, C. 2025. Calibrating Large Language Models with Sample Consistency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(18):19260–19268. AAAI Press.

[Paruchuri et al. 2024] Paruchuri, A.; Garrison, J.; Liao, S.; Hernandez, J.; Sunshine, J.; Althoff, T.; Liu, X.; and McDuff, D. 2024. What Are the Odds? Language Models Are Capable of Probabilistic Reasoning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Miami, Florida, USA: Association for Computational Linguistics.