Recovering Event Probabilities from Large Language Model Embeddings via Axiomatic Constraints

Jian-Qiao Zhu (zhujq@hku.hk)[1,2], Haijiang Yan (haijiang.yan@warwick.ac.uk)[3] & Thomas L. Griffiths (tomg@princeton.edu)[2,4]

[1]Department of Psychology, The University of Hong Kong
[2]Department of Computer Science, Princeton University
[3]Department of Psychology, University of Warwick
[4]Department of Psychology, Princeton University

Abstract

Rational decision-making under uncertainty requires coherent degrees of belief in events. However, event probabilities generated by Large Language Models (LLMs) have been shown to exhibit incoherence, violating the axioms of probability theory. This raises the question of whether coherent event probabilities can be recovered from the embeddings used by the models. If so, those derived probabilities could be used as more accurate estimates in events involving uncertainty, and the embeddings would become more interpretable. To explore this question, we propose enforcing axiomatic constraints, such as the additive rule of probability theory, in the latent space learned by an extended variational autoencoder (VAE) applied to LLM embeddings. This approach enables event probabilities to naturally emerge in the latent space as the VAE learns to both reconstruct the original embeddings and predict the embeddings of semantically related events. We evaluate our method on complementary events (i.e., event A and its complement, event not-A), where the true probabilities of the two events must sum to 1. Experiment results on open-weight language models demonstrate that probabilities recovered from embeddings exhibit greater coherence than those directly generated by the models. Moreover, the recovered probabilities align closely with the true probabilities, while the latent space of the VAE provides interpretable structures.

Keywords: Coherence; Probability Judgments; Large Language Models; Variational Autoencoeder; Axiomatic Constraints

Introduction

Understanding how language models represent and process information through their embeddings is a key focus in modern interpretability research (Elhage et al., 2021; Gurnee et al., 2023; Templeton et al., 2024). Of particular interest are features related to event probabilities, as these shape how models reason and make decisions when faced with uncertainty (Chen et al., 2018; Jia et al., 2024; Requeima et al., 2024; Zheng et al., 2023). Emerging evidence suggests that event probabilities generated by LLMs in text responses (which we denote $\mathbf{P}_{\text{judged}}$) often lack coherence (Zhu & Griffiths, 2024). For instance, when separately prompted about an event A and the complementary event not-A, the LLM-judged probabilities for A and not-A typically do not sum to 1 (i.e., $\mathbf{P}_{\text{judged}}(A) + \mathbf{P}_{\text{judged}}(\neg A) \neq 1$). In contrast, true probabilities inherently satisfy the axiomatic constraints dictated by the rules of probability theory, ensuring that they are always coherent. In addition to the rules of probability theory, LLMs have also been found to violate other axioms (Alsagheer et al., 2024; Horton, 2023; Liu et al., 2024; Raman et al., 2024; Zhu & Griffiths, 2024).

Incoherence can have significant and potentially dangerous consequences. For instance, consider the risks of relying on an incoherent LLM for financial advice or allowing it to make investment decisions on one's behalf. Incoherent beliefs, such as assigning probabilities like $P(\text{price up}) = P(\text{price down}) = 0.6$, can be easily exploited by arbitrageurs. Holding such incoherent beliefs may become "money pumps," as inconsistent judgments create opportunities for others to profit at their expense (Takayanagi et al., 2025; Yang et al., 2020). This phenomenon is explored in Dutch Book arguments, which emphasize the necessity of coherent beliefs to avoid such vulnerabilities (Pettigrew, 2020). Beyond finance, similarly adverse outcomes can arise when incoherent LLMs are applied to critical domains such as healthcare and law.

While the probabilities produced in LLM responses may lack coherence, LLMs are generally capable of generating coherent text descriptions of complementary events (Brown et al., 2020). This suggests that the probabilities produced in LLM outputs may not directly correspond to the probabilities represented within their embeddings. Much as people might not always explicitly articulate their true thoughts, it is possible that LLMs do not generate probabilities that faithfully reflect their internal representations. In this paper, we investigate whether valid event probabilities can be recovered from LLM embeddings by imposing axiomatic constraints designed to enforce coherence, and whether the resulting coherence-imposed embeddings provide greater interpretability than the original embeddings.

Imposing axiomatic constraints on the latent space may initially appear challenging, as these constraints often involve complex relational dependencies among behaviors. However, we demonstrate that under mild assumptions, enforcing the additive rule for complementary events can be achieved by enforcing structure in the latent space, akin to learning disentangled latent representations (Bengio, 2013; Chen et al., 2018; Mathieu et al., 2019). In this work, we make three key contributions. First, we introduce a novel unsupervised learning approach for extracting an interpretable latent space from LLM embeddings by enforcing axiomatic constraints. Second, we show that a subset of the latent variables, constrained by the axioms of probability theory, corresponds to

In G. Boleda, I. Dautriche, M. Maldonado, C. Saldana and T. Torrent (Eds.), *Proceedings of the 48th Annual Conference of the Cognitive Science Society*

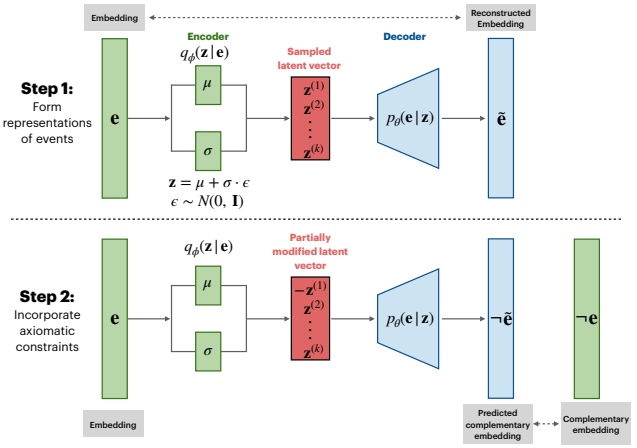

Figure 1: A schematic illustration of the two-step VAE-based learning algorithm. The objective is to distill axiomatic constraints between embeddings into the modified latent variables. **(top)** In the first step, a standard VAE is used to disentangle explanatory factors within the latent space from the model embeddings. This involves learning to reconstruct the original LLM embeddings, $\mathbf{e}$, by encoding them into a compressed latent space, $\mathbf{z}$, and subsequently decoding reconstructed embeddings, $\tilde{\mathbf{e}}$. The latent space is modeled as a multivariate Gaussian distribution with mean μ and standard deviation σ. **(bottom)** In the second step, a subset of the latent variables is modified after encoding, and the modified latent vector is passed through the decoder to predict the embeddings of complementary events, $\neg\tilde{\mathbf{e}}$. The probabilistic encoder and probabilistic decoder are denoted $q_\phi(\mathbf{z}|\mathbf{e})$ and $p_\theta(\mathbf{e}|\mathbf{z})$ respectively. Owing to the symmetry property (i.e., $\neg(\neg\mathbf{e}) = \mathbf{e}$), both $\mathbf{e}$ and $\neg\mathbf{e}$ can be used as input embeddings.

event probabilities implicitly encoded within LLM embeddings. Finally, we demonstrate that the recovered probabilities exhibit improved coherence and align closely with the true probabilities than those directly derived from the original embeddings.

Problem Formulation

Let us consider a dataset of N prompts, each representing an event of interest, denoted as $\mathbf{S} = \{\mathbf{s}^{(i)}\}_{i=1}^{N}$. For example, a prompt might be $\mathbf{s}$: "*What is the probability of rolling a 5 on a 6-sided die?*" Our goal is to investigate how LLMs represent event probabilities, relying solely on model embeddings and without access to the true probabilities of the events. Specifically, we aim to recover the latent representation of the event probability, $\mathbf{z}^{(i)}$, by extracting the corresponding embedding $\mathbf{e}^{(i)}$ from LLM, resulting in a dataset of embeddings: $\mathbf{E} = \{\mathbf{e}^{(i)}\}_{i=1}^{N}$.

Importantly, for each prompt, we also have a prompt for the complementary event: $\neg\mathbf{S} = \{\neg\mathbf{s}^{(i)}\}_{i=1}^{N}$. For the dice example above, the prompt for the complementary event would be $\neg\mathbf{s}$: "*What is the probability of rolling a number other than 5 on a 6-sided die?*" Similarly, for this dataset of complementary events, we extract embeddings from the LLMs, forming a dataset of complementary event embeddings: $\neg\mathbf{E} = \{\neg\mathbf{e}^{(i)}\}_{i=1}^{N}$.

Within this formulation, we aim to address three interrelated problems. First, we seek to understand how event probabilities are encoded within LLM embeddings. Second, we develop efficient methods to recover and potentially enhance these probability representations. Third, we explore the implications of the learned representations when axiomatic constraints are correctly imposed, examining how such constraints shape both their coherence and interpretability (see Appendix B for related work[1]).

Method

Our unsupervised approach aims to learn a set of disentangled latent variables that serve two key purposes. First, these latent variables can be used to reconstruct the original embeddings. Second, through targeted modifications of specific latent variables, they can also be used to accurately predict the embeddings of complementary events. These modifications effectively encode the relationship between complementary events' embeddings–by activating or deactivating the modification, the same latent variables can predict either embedding. We can then enforce desired axiomatic constraints that respect the relationship between complementary events by imposing a specific prior structure on these modified latent variables. Conceptually, our approach aligns with methods that constrain neural networks to exploit the symmetry of problems through enforcing equivariance (Cohen & Welling, 2016; Zaheer et al., 2017).

Specifically, given our focus on event probabilities from LLM embeddings $\mathbf{e} \in \mathbb{R}^d$, all other information within the embedding is considered extraneous and hence should be effectively compressed. To achieve this, we aim to map the original embedding into a smaller, more compact latent space $\mathbf{z} \in \mathbb{R}^k$, where $k \ll d$. Ideally, for simplicity and interpretability, the event probability would be encapsulated in a single latent variable (e.g., $\mathbf{z}^{(1)}$), with the remaining latent variables representing information unrelated to the event probability.

Using two datasets of model embeddings for complementary events, $\mathbf{E}$ and $\neg\mathbf{E}$, we employ the autoencoding variational Bayes algorithm to learn compressed and disentangled latent representations $\mathbf{z}$. As illustrated in Figure 1 and Algorithm 1, the proposed method consists of two interleaved steps during training.

In the first step, we adopt the standard variational autoencoder (VAE) framework, where two neural networks separately implement the probabilistic encoder $q_\phi(\mathbf{z}|\mathbf{e})$ and decoder $p_\theta(\mathbf{e}|\mathbf{z})$ (Kingma & Welling, 2014). The parameters of these networks (ϕ and θ) are jointly optimized to maximize the likelihood of the embeddings, $p(\mathbf{e})$. The prior distribution over latent variables is assumed to be a centered isotropic multivariate Gaussian, $p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, \mathbf{I})$.

[1]The Appendix is available at the following link: https://osf.io/a3vw8

In the second step, a key deviation from the standard VAE framework is introduced to capture the constraints that the axioms of probability theory impose on complementary events: this step optimizes the encoder and decoder parameters to maximize the probability of the complementary embeddings conditioned on the original embeddings, $p(\neg\mathbf{e}|\mathbf{e})$. The approach assumes that the constraint between $\neg\mathbf{e}$ and $\mathbf{e}$ can be captured by selectively modifying a subset of latent variables after encoding. Because the constraint is explicitly incorporated during the latent modifications, the encoder-decoder networks from Step 1 can be reused. Combined with the remaining unmodified latent variables, the partially modified latent vector is then passed through the decoder network to predict the embedding of the complementary event.

In summary, if the trained model successfully reconstructs the input embedding from a compressed latent representation and predicts the complementary embedding by partially modifying the same latent representation, the latent space should capture interpretable and meaningful low-dimensional information. Furthermore, the subset of latent variables that undergo modification should adhere to the constraints imposed during the modification process.

Variational bounds. Our learning algorithm aims to simultaneously capture two probability distributions: $p(\mathbf{e})$ and $p(\neg\mathbf{e}|\mathbf{e})$. We assume that LLM embeddings are generated through a random process involving latent variables $\mathbf{z}$. Approximating the intractable marginal likelihood $p(\mathbf{e})$ is achieved by jointly optimizing an encoder, $q_\phi(\mathbf{z}|\mathbf{e})$, and a decoder, $p_\theta(\mathbf{e}|\mathbf{z})$. Specifically, the optimization focuses on a variational lower bound of the marginal likelihood:

$$\begin{aligned}\log p(\mathbf{e}) &\geq \mathcal{L}_1(\phi,\theta;\mathbf{e}) \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{e})}\Big[\log p_\theta(\mathbf{e}|\mathbf{z})\Big] - D_{KL}\Big(q_\phi(\mathbf{z}|\mathbf{e}) \parallel p_\theta(\mathbf{z})\Big)\end{aligned} \tag{1}$$

The encoder and decoder can be parameterized using two neural networks (i.e., ϕ and θ). The variational lower bound is differentiable with respect to these neural network parameters, enabling gradient-based optimization. To facilitate training, the reparameterization trick can be applied (Kingma & Welling, 2014).

Similarly, the variational bound for $p(\mathbf{e})$ can be extended to $p(\neg\mathbf{e}|\mathbf{e})$ (see Appendix A):

$$\begin{aligned}\log p(\neg\mathbf{e}|\mathbf{e}) \geq\ & \mathbb{E}_{q(\mathbf{z}|\mathbf{e})}\Big[\log p(\neg\mathbf{e}|\mathbf{z},\mathbf{e})\Big] \\ & - D_{KL}\Big(q(\mathbf{z}|\mathbf{e}) \parallel p(\mathbf{z}|\mathbf{e})\Big)\end{aligned} \tag{2}$$

This new bound introduces two additional quantities that are intractable. The generative distribution, $p(\neg\mathbf{e}|\mathbf{z},\mathbf{e})$, can be simplified to depend solely on the latent variable. This assumes that the modifications in the latent space are sufficient to generate $\neg\mathbf{e}$:

$$p(\neg\mathbf{e}|\mathbf{z},\mathbf{e}) \approx p(\neg\mathbf{e}|T(\mathbf{z})) \tag{3}$$

where the modification operation consists of negating the first latent variable while preserving all others: $T(\mathbf{z}) = [-\mathbf{z}^{(1)}, \mathbf{z}^{(-1)}]$.

Furthermore, the true posterior $p(\mathbf{z}|\mathbf{e})$ can be approximated using the posterior learned during the optimization of the variational bounds for $p(\mathbf{e})$:

$$p(\mathbf{z}|\mathbf{e}) \approx q_\phi(\mathbf{z}|\mathbf{e}) \tag{4}$$

Therefore, the training objective in the second step can be rewritten as:

$$\begin{aligned}\mathcal{L}_2(\phi,\theta;\mathbf{e},\neg\mathbf{e},\phi_0) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{e})}\Big[& \log p_\theta(\neg\mathbf{e}| - \mathbf{z}^{(1)}, \mathbf{z}^{(-1)})\Big] \\ & - D_{KL}\Big(q_\phi(\mathbf{z}|\mathbf{e}) \parallel q_{\phi_0}(\mathbf{z}|\mathbf{e}))\Big)\end{aligned} \tag{5}$$

where ϕ_0 denotes the parameters of the probabilistic encoder learned from the first step.

In short, we hypothesize that the relationship between embeddings for complementary events (i.e., the relationship between $\mathbf{e}$ and $\neg\mathbf{e}$) is captured in the first latent variable through the sign-flipping operation (i.e., the modification between $\mathbf{z}^{(1)}$ and $-\mathbf{z}^{(1)}$). This implies that the probabilities of the two complementary events are intrinsically linked to the sign-flipping transformation of the first latent variable. Consequently, the first latent variable is best interpreted as representing log odds, rather than raw probabilities. Finally, the log odds recorded in the modified latent variable, $\mathbf{z}^{(1)}$, are converted back to the probability scale:

$$\mathbf{P}_{\text{recovered}} = \frac{e^{\mathbf{z}^{(1)}}}{1 + e^{\mathbf{z}^{(1)}}} \tag{6}$$

By flipping the sign of $\mathbf{z}^{(1)}$, we enforce that the recovered probabilities satisfy the additive rule where the probabilities of complementary events must sum to 1:

$$\mathbf{P}_{\text{recovered}} + (1 - \mathbf{P}_{\text{recovered}}) = \frac{e^{\mathbf{z}^{(1)}}}{1 + e^{\mathbf{z}^{(1)}}} + \frac{e^{-\mathbf{z}^{(1)}}}{1 + e^{-\mathbf{z}^{(1)}}} \tag{7}$$

Experiments

We now demonstrate the effectiveness of our proposed method using a dataset of dice-related events. The primary advantage of using a probabilistic device like dice, as opposed to more ambiguous or real-world events, lies in the availability of their true probabilities. When the underlying true probabilities are known, the accuracy of probability estimates can be assessed by comparing them against these true values. This enables us to evaluate not only the coherence among a set of related probability estimates but also their alignment with the true probabilities.

We simulated all possible outcomes from rolling multi-sided dice, including scenarios such as a single roll of a 6-sided die, two rolls of a 6-sided die, and five rolls of a 4-sided die. In addition to determining the probability of observing a specific number or sum from the rolls, we also considered comparative queries, such as whether the sum is less than or greater than a given number (e.g., "*What is the probability that the sum of 8 rolls of a 10-sided die is greater than 15?*"). In total, we generated $N = 1,728$ probability questions along with their corresponding complementary events.

Moreover, we constructed a held-out test set consisting of 480 dice events. These events were generated using the same procedure as the training set but varied in the number of rolls and the number of sides on the die. Importantly, none of the 480 test events or their complements appeared in the training set. This test set constitutes approximately 28% of the size of the training set (see Appendix D for details).

We primarily focused on the Gemma-2-9b-instruct model, chosen for its open-weight architecture and robust performance on established benchmarks (Gemma Team et al., 2024). We replicated our experiments using Llama-3.1-8b-instruct (Dubey et al., 2024), with detailed results in Appendix K.

Judged probabilities. We first evaluate the intuitive probability judgments generated in text by the Gemma model. To facilitate comparison with the probabilities recovered from model embeddings, we include the following system prompt before each probability question: "*You are asked to only provide an intuitive probability judgment as a number between 0 and 1.*" The text responses were filtered to retain only a single real number between 0 and 1 representing the model's judged probability. The model's temperature was set to 1.

For complementary events, we define an incoherence measure by comparing the sum of the probabilities of the two events to the expected coherent sum of 1:

$$\text{Incoherence} = |\mathbf{P} + \neg\mathbf{P} - 1| \tag{8}$$

This absolute difference quantifies the extent of incoherence in the probability estimates.

As shown in Table 1 and Figure 2a, $\mathbf{P}_{\text{judged}}$ exhibits mean incoherence scores of 0.13 on the training set and 0.14 on the test set, indicating a substantial deviation from the perfect coherence exhibited by $\mathbf{P}_{\text{true}}$. These findings replicate the results reported in (Zhu & Griffiths, 2024). Despite this incoherence, $\mathbf{P}_{\text{judged}}$ demonstrates good alignment with the true probabilities, with Pearson's $r = 0.80$ ($p < .01$) and $r = 0.73$ ($p < .01$) for the training and test sets, respectively.

To improve coherence, we also normalize $\mathbf{P}_{\text{judged}}$ by taking pairs of judgments for complementary events and dividing each by their summed total. This normalization imposes the arithmetic constraint and therefore reduces incoherence to zero, while leaving accuracy, measured by MSE, relatively unchanged ($t(3455) = -2.54$, $p = .01$ for the training set and $t(959) = -1.71$, $p = .09$ for the test set). Furthermore, we explored an alternative approach in which complementary events were jointly presented within the same prompt, explicitly allowing the model to reason before providing its final probability judgments. This method significantly improved both the accuracy and coherence of probability judgments (see Appendix E).

Recovered probabilities. Now we apply unsupervised methods on LLM embeddings in order to learn to recover interpretable event probabilities. For each probability question, we extracted the embedding of the last token from the final layer before the logit computation, leading to two datasets of $\{\mathbf{e}^{(i)}\}_{i=1}^{1728}$ and $\{\neg\mathbf{e}^{(i)}\}_{i=1}^{1728}$. Notably, neither $\mathbf{P}_{\text{true}}$ nor $\mathbf{P}_{\text{judged}}$ is provided during this process. Due to the symmetry property (i.e., $\neg(\neg\mathbf{e}) = \mathbf{e}$), both $\mathbf{e}$ and $\neg\mathbf{e}$ were used as input embeddings for the VAE.

Since the modified latent variable is interpreted as representing log odds, the centered isotropic Gaussian prior $p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, \mathbf{I})$ effectively enforces the axiomatic constraint of complementary events. This is because the additive rule for complementary events in the log-odds space corresponds to the sum of the log odds of the two events equaling zero. Therefore, the Gaussian prior with a mean of zero can be interpreted as a form of regularization that aligns the latent variables with the imposed axiomatic constraint. As a corollary, stricter regularization can be imposed by increasing the weight of the divergence from this prior, more strongly enforcing adherence to the axiomatic constraint.

In practice, we employ β-VAE (Higgins et al., 2017) to control the regularization strength toward the desired axiomatic constraint. β-VAE introduces a hyperparameter, β, which scales the KL divergence term in the VAE objective (see Equation 1 and 2). In our experiments, we set $\beta = 5$ to emphasize regularization toward the Gaussian prior. When converting log odds back to the probability scale, we applied the same value of 5 for temperature. Step 1 and Step 2 of the training process were interleaved every 10 training episodes. In each episode, a batch of 128 embeddings, along with those of the complementary events, was randomly selected. AdamW with a learning rate of 1e-4 was used for optimization (Loshchilov, 2017).

The architecture consists of an encoder and decoder, each implemented as a 3-layer feedforward neural network with GELU activation functions (Hendrycks & Gimpel, 2016). Each layer contains d neurons, where d is the embedding dimension, and the hidden dimension is set to 10.

Figure S2 presents the mean values of the 10 latent variables after training, comparing the embeddings for the two complementary events. All latent variables exhibit a positive relationship, except for the modified latent variable (i.e., $\mathbf{z}^{(1)}$). This suggests that the relationship of complementary events between $\mathbf{e}$ and $\neg\mathbf{e}$ has been successfully distilled into this specific latent variable. As illustrated in Figure 2b, the log odds values of $\mathbf{z}^{(1)}$ were transformed back into the probability scale, yielding $\mathbf{P}_{\text{recovered}}$ (see Equation 6).

On the held-out test set (see Table 1), we find that $\mathbf{P}_{\text{recovered}}$ is significantly more coherent than $\mathbf{P}_{\text{judged}}$, as indicated by a lower incoherence score ($t(479) = -10.47, p < .01$). However, its coherence is statistically indistinguishable from that of the normalized $\mathbf{P}_{\text{judged}}$ ($t(479) = 1.01, p = .31$). In terms of accuracy, measured by MSE, $\mathbf{P}_{\text{recovered}}$ performs comparably to $\mathbf{P}_{\text{judged}}$ ($t(959) = -1.58, p = .12$). Additionally, $\mathbf{P}_{\text{recovered}}$ and $\mathbf{P}_{\text{judged}}$ are strongly correlated (Pearson's $r = 0.90, p < .01$), indicating substantial alignment between the two (see Appendix F). Taken together, these results suggest that while LLM embeddings encode information that supports more coherent probability estimates, there remains considerable room for accuracy improvement in how event probabilities are represented.

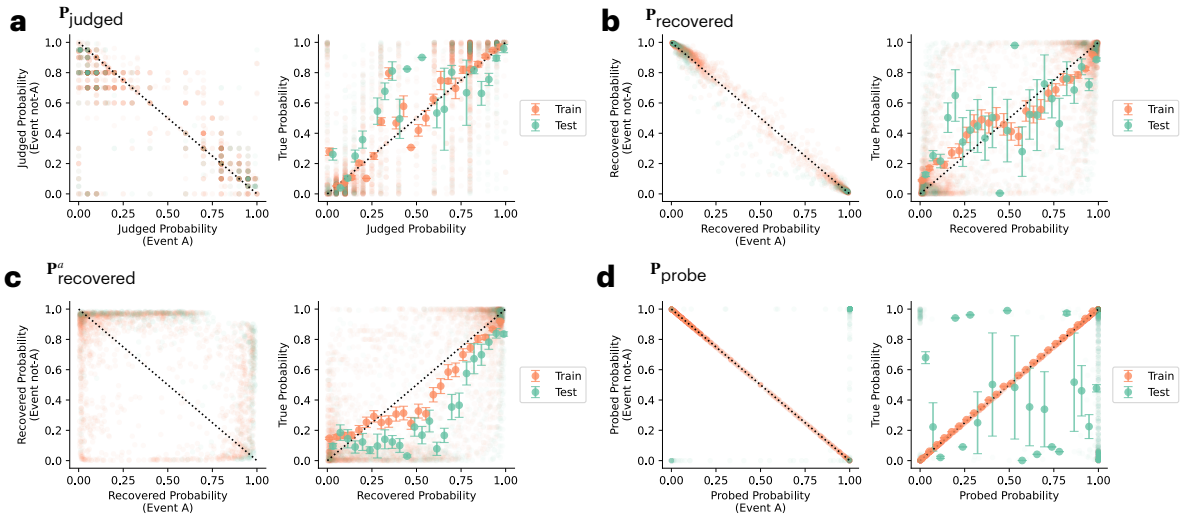

Figure 2: Comparison of event probability estimates elicited from Gemma-2-9b-instruct: (**a**) judged probabilities, (**b**) recovered probabilities, (**c**) recovered probabilities with Step 2 ablated, and (**d**) probabilities predicted by a linear probe. As indicated by the dotted black lines in the left panel, coherent probability estimates for event A and its complement (not-A) should sum to 1. The right panel compares the elicited probabilities with the true probabilities. Error bars represent standard errors of the mean for the window-binned scatter points.

Table 1: Quality of probability estimates (Gemma-2-9b-instruct). Incoherence is quantified using Equation 8, while accuracy is assessed via Pearson's correlation coefficient (r) and mean squared error (MSE) with respect to the true probabilities. Models that use LLM embeddings to predict event probabilities are trained exclusively on the training set, with model weights held fixed during evaluation on the test set.

Dataset	Elicitation method	Incoherence ↓ (95% CI)	Pearson's r ↑	MSE ↓
Train	$\mathbf{P}_{\text{judged}}$	0.1297 ([0.1218, 0.1376])	0.8032 ($p < .01$)	0.0601
	$\mathbf{P}_{\text{judged}}$ (normalized)	0.0023 ([0.0000, 0.0046])	0.7862 ($p < .01$)	0.0654
	$\mathbf{P}_{\text{recovered}}$	0.0227 ([0.0211, 0.0243])	0.8264 ($p < .01$)	0.0587
	$\mathbf{P}^{a}_{\text{recovered}}$	0.2948 ([0.2827, 0.3069])	0.7610 ($p < .01$)	0.0750
	$\mathbf{P}_{\text{probe}}$	0.0006 ([0.0006, 0.0007])	1.0 ($p < .01$)	< 0.0001
Test	$\mathbf{P}_{\text{judged}}$	0.1366 ([0.1190, 0.1542])	0.7286 ($p < .01$)	**0.0927**
	$\mathbf{P}_{\text{judged}}$ (normalized)	**0.0021** ([-0.0020, 0.0062])	0.7091($p < .01$)	**0.1007**
	$\mathbf{P}_{\text{recovered}}$	0.0383 ([0.0314, 0.0452])	**0.7328** ($p < .01$)	**0.1014**
	$\mathbf{P}^{a}_{\text{recovered}}$	0.3457 ([0.3223, 0.3692])	0.7167 ($p < .01$)	0.1140
	$\mathbf{P}_{\text{probe}}$	0.8535 ([0.8241, 0.8829])	-0.1328 ($p < .01$)	0.4654

Note. $\mathbf{P}^{a}_{\text{recovered}}$ refers to the recovered event probabilities from the ablated model. $\mathbf{P}_{\text{probe}}$ refers to the probabilities predicted by a linear probe. Bolded numbers indicate the best-performing methods on the test set. When multiple methods are highlighted, their performance differences are not statistically significant at the 0.05 level.

Non-modified latent variables. We examine the remaining nine non-modified latent variables, $\mathbf{z}^{(2:10)}$, which exhibited a positive relationship between $\mathbf{e}$ and $\neg\mathbf{e}$. To assess whether these latent variables are interpretable, we regress the features of the prompts onto the mean values of these latent variables. For each prompt, we identified six features, as detailed in Table S1 features column. To illustrate, consider the prompt "*What is the probability of rolling a 5 on a 6-sided die?*". This prompt can be characterized by the following six features: (i) one roll, (ii) a 6-sided die, (iii) an outcome of 5, (iv) a true probability of 1/6, (v) no reference to a sum, and (vi) a comparison type of "equal to."

Using Lasso regression with a penalty ratio of 1, we find that many non-modified latent variables are associated with specific features of the prompts. As shown in Table S1 (see Appendix G), $\mathbf{z}^{(9)}$ is strongly associated with the number of rolls, accounting for 50% variance (R^2 = 50%); $\mathbf{z}^{(7)}$ corresponds to whether a prompt involves a summation of multiple rolls (R^2 = 61%); and $\mathbf{z}^{(5)}$ is linked to the comparison type (R^2 = 61%). In addition, true probabilities and outcomes

appear to be influenced by a combination of latent variables, with the true probabilities being predominantly affected by the modified latent variable, $\mathbf{z}^{(1)}$. While the modified $\mathbf{z}^{(1)}$ is mostly correlated with the true probabilities, certain other latent variables, such as $\mathbf{z}^{(5,7,9)}$, exhibit correlations with multiple features. This suggests the presence of *polysemanticity* (Bolukbasi et al., 2021; Elhage et al., 2022), a phenomenon where neuron activates for a range of different features, in the compressed latent space.

Ablation study. To evaluate the effectiveness of our proposed two-step method, we conducted an ablation study. A key concern is whether Step 2 in Figure 1, which modifies the latent vector for more targeted isolation in the latent space, is necessary to identify latent variables associated with event probabilities. Specifically, we sought to determine whether a disentangled representation could be achieved by applying a standard VAE to LLM embeddings without Step 2.

To address this question, we removed the interleaved training from our method and directly implemented a β-VAE using the same hyperparameters and architecture. In this ablated model, the objective is learning to reconstruct the input embeddings. Notably, both $\mathbf{e}$ and $\neg\mathbf{e}$ were used as input embeddings, ensuring that the ablated model was trained on the same amount of data as before. The learned latent representations are shown in Figure S3 (see Appendix I). These latent representations do not exhibit clear relationships (positive or negative) between $\mathbf{e}$ and $\neg\mathbf{e}$. This suggests that the phenomenon of polysemanticity is more pronounced in representations learned from the standard β-VAE compared to those produced by our two-step method. Consequently, it is challenging to identify latent variables in the ablated model that are specifically associated with event probabilities. While incorporating additional supervision signals from true probabilities could potentially enable the construction of a linear model that combines multiple latent variables to predict event probabilities, this would undermine the objective of recovering event probabilities through unsupervised learning.

Nonetheless, the latent variable associated with event probabilities is expected to exhibit the strongest negative correlation between complementary events. Consistent with this expectation, we found the 9th latent variable most negatively correlated (Pearson's $r = -0.31$, $p < .01$). The recovered probabilities from the ablated model (which we denote $\mathbf{P}^{a}_{\text{recovered}}$) were derived based on this latent variable (see Figure 2c). However, $\mathbf{P}^{a}_{\text{recovered}}$ are less coherent than $\mathbf{P}_{\text{recovered}}$ ($t(1727) = -44.68$, $p < .01$ for incoherence scores) and also less accurate ($t(3455) = -6.71$, $p < .01$ for MSE) in the training set. These findings suggest that ablating Step 2 results in a less interpretable latent space.

Linear Probe. Finally, we train a linear probe to map LLM embeddings onto the log-odds of the true probabilities. Specifically, $\text{logit}(\mathbf{P}_{\text{true}}) = \beta^{\top}\mathbf{e}$, where β denotes the regression coefficients corresponding to the embedding dimension. At test time, these coefficients are applied to embeddings from held-out events, and the predicted logits are transformed back to probabilities. As shown in Figure 2d and Table 1, the probe achieves near-perfect coherence and accuracy on the training set but fails to generalize to the test set. In Appendix J, we examine an alternative approach using $\mathbf{P}_{\text{true}}$ directly as the training target. While this yields a more stable mapping from embeddings to probabilities, the resulting predictions on the test set are not guaranteed to be coherent, as they may fall outside the [0, 1] interval. These findings from the linear probe suggest that event probabilities are likely embedded in a nonlinear manner.

Discussion

Our unsupervised approach recovers more coherent and interpretable event probabilities from LLM embeddings than those obtained through prompting probability judgments in text. This improvement is achieved by enforcing the additive rule of probability theory on the embeddings of complementary events. Moreover, the resulting latent space exhibits high interpretability, with individual latent variables corresponding to specific features of the prompt. This suggests that when properly trained, the compressed latent representation derived from LLM embeddings can effectively disentangle LLM embeddings without introducing sparsity bias on the latent space.

Limitations and Future Directions. We note several limitations of our current work that suggest promising directions for future research. The latent space can be further extended to accommodate additional axiomatic constraints. For instance, the relationship between conjunctive and constituent events could be enforced through Bayes' rule: $P(A \text{ and } B) = P(A)P(B|A)$. Future work could explore methods for effectively implementing such axiomatic constraints. Likewise, generalizing the method to capture more naturalistic events with real-world relevance is an important research direction.

Moreover, a natural next step is to investigate whether editing neuron activations based on the learned latent variables can produce more coherent probability judgments. We consider this a promising direction for establishing a clearer causal relationship. Our approach could be extended to explore various other constraints across different LLM modalities. For instance, in multimodal LLMs, embeddings should exhibit rotational invariance when processing images of the same object from different angles. Future research could investigate how to impose such constraints.

Conclusion. We have shown that coherent probability judgments can be extracted from LLMs by imposing axiomatic constraints on the underlying representations. The successful recovery of coherent probability estimates from LLM embeddings suggests that critical information might be lost when LLMs generate probability judgments in text form. Our findings support recent hypotheses that certain LLM capabilities – in this case, the ability to generate more accurate probability judgments – remain "hidden" when only examining model responses but can be unlocked through self-improvement techniques such as bootstrapping (Zelikman et al., 2022).

Acknowledgments

This work and related results were made possible with the support of the NOMIS Foundation. J.-Q. Zhu acknowledges support from the HKU-100 scheme of the University of Hong Kong. H. Yan acknowledges the Chancellor's International Scholarship from the University of Warwick for additional support.

References

Alsagheer, D., Karanjai, R., Diallo, N., Shi, W., Lu, Y., Beydoun, S., & Zhang, Q. (2024). Comparing rationality between large language models and humans: Insights and open questions. *arXiv preprint arXiv:2403.09798*.

Bengio, Y. (2013). Deep learning of representations: Looking forward. *International Conference on Statistical Language and Speech Processing*, 1–37.

Bolukbasi, T., Pearce, A., Yuan, A., Coenen, A., Reif, E., Viégas, F., & Wattenberg, M. (2021). An interpretability illusion for bert. *arXiv preprint arXiv:2104.07143*.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, *33*, 1877–1901.

Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., & Lerchner, A. (2018). Understanding disentangling in β-VAE. *arXiv preprint arXiv:1804.03599*.

Chen, R. T., Li, X., Grosse, R. B., & Duvenaud, D. K. (2018). Isolating sources of disentanglement in variational autoencoders. *Advances in Neural Information Processing Systems*, *31*.

Cohen, T., & Welling, M. (2016). Group equivariant convolutional networks. *International Conference on Machine Learning*, 2990–2999.

Cunningham, H., Ewart, A., Riggs, L., Huben, R., & Sharkey, L. (2023). Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.

Demircan, C., Saanum, T., Jagadish, A. K., Binz, M., & Schulz, E. (2024). Sparse autoencoders reveal temporal difference learning in large language models. *arXiv preprint arXiv:2410.01280*.

Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Yang, A., Fan, A., et al. (2024). The LLaMA 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., et al. (2022). Toy models of superposition. *arXiv preprint arXiv:2209.10652*.

Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., et al. (2021). A mathematical framework for transformer circuits. *Transformer Circuits Thread*, *1*(1), 12.

Gemma Team, n., Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S., Sifre, L., Rivière, M., Kale, M. S., Love, J., et al. (2024). Gemma: Open models based on Gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. *International conference on machine learning*, 1321–1330.

Gurnee, W., Nanda, N., Pauly, M., Harvey, K., Troitskii, D., & Bertsimas, D. (2023). Finding neurons in a haystack: Case studies with sparse probing. *arXiv preprint arXiv:2305.01610*.

Hendrycks, D., & Gimpel, K. (2016). Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Higgins, I., Matthey, L., Pal, A., Burgess, C. P., Glorot, X., Botvinick, M. M., Mohamed, S., & Lerchner, A. (2017). Beta-VAE: Learning basic visual concepts with a constrained variational framework. *International Conference on Learning Representations*, *3*.

Horton, J. J. (2023). *Large language models as simulated economic agents: What can we learn from homo silicus?* (Tech. rep.). National Bureau of Economic Research.

Hyvärinen, A., Khemakhem, I., & Morioka, H. (2023). Nonlinear independent component analysis for principled disentanglement in unsupervised deep learning. *Patterns*, *4*(10).

Jia, J., Yuan, Z., Pan, J., McNamara, P. E., & Chen, D. (2024). Decision-making behavior evaluation framework for LLMs under uncertain context. *arXiv preprint arXiv:2406.05972*.

Kim, H., & Mnih, A. (2018). Disentangling by factorising. *International conference on machine learning*, 2649–2658.

Kingma, D. P., & Welling, M. (2014). Auto-Encoding Variational Bayes. *International Conference on Learning Representations*.

Kolmogorov, A. N., & Bharucha-Reid, A. T. (2018). *Foundations of the theory of probability: Second english edition*. Courier Dover Publications.

Kumar, A., Liang, P. S., & Ma, T. (2019). Verified uncertainty calibration. *Advances in Neural Information Processing Systems*, *32*.

Liu, R., Geng, J., Peterson, J. C., Sucholutsky, I., & Griffiths, T. L. (2024). Large language models assume people are more rational than we really are. *arXiv preprint arXiv:2406.17055*.

Loshchilov, I. (2017). Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Mathieu, E., Rainforth, T., Siddharth, N., & Teh, Y. W. (2019). Disentangling disentanglement in variational autoencoders. *International Conference on Machine Learning*, 4402–4412.

Ng, A., et al. (2011). Sparse autoencoder. *CS294A Lecture notes*, *72*(2011), 1–19.

Pettigrew, R. (2016). *Accuracy and the laws of credence*. Oxford University Press.

Pettigrew, R. (2020). *Dutch book arguments*. Cambridge University Press.

Raman, N., Lundy, T., Amouyal, S., Levine, Y., Leyton-Brown, K., & Tennenholtz, M. (2024). Rationality report cards: Assessing the economic rationality of large language models. *arXiv preprint arXiv:2402.09552*.

Requeima, J., Bronskill, J., Choi, D., Turner, R. E., & Duvenaud, D. (2024). Llm processes: Numerical predictive distributions conditioned on natural language. *arXiv preprint arXiv:2405.12856*.

Shrivastava, V., Liang, P., & Kumar, A. (2023). Llamas know what gpts don't show: Surrogate models for confidence estimation. *arXiv preprint arXiv:2311.08877*.

Takayanagi, T., Takamura, H., Izumi, K., & Chen, C.-C. (2025). Can GPT-4 sway experts' investment decisions? *Findings of the Association for Computational Linguistics: NAACL 2025*, 374–383.

Templeton, A., Conerly, T., Marcus, J., Lindsey, J., Bricken, T., Chen, B., Pearce, A., Citro, C., Ameisen, E., Jones, A., et al. (2024). Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. transformer circuits thread.

Wang, K., Variengien, A., Conmy, A., Shlegeris, B., & Steinhardt, J. (2022). Interpretability in the wild: A circuit for indirect object identification in GPT-2 small. *arXiv preprint arXiv:2211.00593*.

Yang, L., Ng, T. L. J., Smyth, B., & Dong, R. (2020). Html: Hierarchical transformer-based multi-task learning for volatility prediction. *Proceedings of The Web Conference 2020*, 441–451.

Ye, F., Yang, M., Pang, J., Wang, L., Wong, D. F., Yilmaz, E., Shi, S., & Tu, Z. (2024). Benchmarking LLMs via uncertainty quantification. *arXiv preprint arXiv:2401.12794*.

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., & Smola, A. J. (2017). Deep sets. *Advances in Neural Information Processing Systems*, *30*.

Zelikman, E., Wu, Y., Mu, J., & Goodman, N. (2022). Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, *35*, 15476–15488.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al. (2023). Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. *Advances in Neural Information Processing Systems*, *36*, 46595–46623.

Zhu, J.-Q., & Griffiths, T. L. (2024). Incoherent probability judgments in large language models. *Proceedings of the Annual Meeting of the Cognitive Science Society*, *46*.

Zhu, J.-Q., Newall, P. W., Sundh, J., Chater, N., & Sanborn, A. N. (2022). Clarifying the relationship between coherence and accuracy in probability judgments. *Cognition*, *223*, 105022.