

A Necessary Step toward Faithfulness: Measuring and Improving Consistency in Free-Text Explanations

Lingjun Zhao

University of Maryland

College Park, Maryland, USA

lzhao123@umd.edu

Hal Daumé III

University of Maryland

College Park, Maryland, USA

hal3@umd.edu

Abstract

Faithful free-text explanations are important to ensure transparency in high-stakes AI decision-making contexts, but they are challenging to generate by language models and assess by humans. In this paper, we present a measure for Prediction-EXplanation (PEX) consistency, by extending the concept of weight of evidence. This measure quantifies how much a free-text explanation supports or opposes a prediction, serving as an important aspect of explanation faithfulness. Our analysis reveals that more than 62% explanations generated by large language models lack this consistency. We show that applying direct preference optimization improves the consistency of generated explanations across three model families, with improvement ranging from 43.1% to 292.3%. Furthermore, we demonstrate that optimizing this consistency measure can improve explanation faithfulness by up to 9.7%.¹

1 Introduction

Explainable AI systems are those that can specify the relationship between an output or prediction and the deductive or nomological process that led to that prediction through logically consistent and empirically grounded means (Woodward and Ross, 2003).² Such explanations are often called “faithful” in the sense that they accurately reflect a model’s true reasoning process (Jacovi and Goldberg, 2020; Lyu et al., 2024), and faithfulness is often seen as crucial for trustworthiness, transparency, and accountability.

Faithfulness, however, is a difficult construct to measure for complex models (such as deep neural networks) precisely because we do not know a

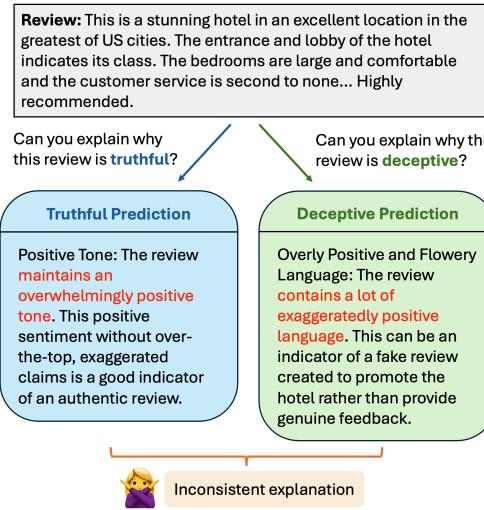


Figure 1: Example of GPT-4 model generating explanations for truthful or deceptive prediction about a hotel review’s authenticity. While each explanation appears plausible on its own, they are *inconsistent*: both the truthful and deceptive explanations rely on the same evidence “use a lot of positive language”.

priori exactly how a model combines its input features to make a prediction, beyond trivially writing out the entire computation—which is then impenetrable to a person. When an explanation is given in the form of natural language, increasingly common especially in the context of large language models (Camburu et al., 2018; Wiegreffe et al., 2022) the challenge increases because humans often conflate faithfulness and plausibility: how convincing it appears (Jacovi and Goldberg, 2020; Lyu et al., 2023; Wiegreffe et al., 2022). This leads to models that produce plausible yet unfaithful accounts of their reasoning processes (Ye and Durrett, 2022; Lanham et al., 2023), which risks eroding the very trustworthiness that they aim to improve.

In this paper, we aim to improve model *faithfulness* by optimizing the *consistency* between an explanation and its explanandum (e.g., a model prediction). Following Miller (2019)’s criterion,

¹ Our code is publicly released at https://github.com/lingjunzhao/PEX_consistency.

² Other forms of explanations exist, such as those that aim to provide useful information for better decision making (Kayser et al., 2024; Han et al., 2023) or debugging (Ribeiro et al., 2016; Lundberg and Lee, 2017).

a necessary condition for faithful explanations is that an explanation should contrastively refute the negation of the explanandum. Therefore, for an explanation to faithfully represent a deductive or nomological process, it cannot simultaneously support a prediction y and its negation $\neg y$. We formalize this notion as *inconsistency*. For example, in Figure 1, a model uses essentially the same explanation (“use a lot of positive language”) to argue in favor of a hotel review being both authentic and deceptive in this opinion spam classification task.

Our work proceeds in three steps. First, we ask how consistent large language model (LLM)-generated explanations are in practice. To do this, we introduce a measure of consistency, PEX (for **P**rediction-**E**Xplanation consistency), which can be applied to models that provide probabilities over their generated outputs. PEX leverages the weight of evidence framework (Melis et al., 2021; Good, 1985) to quantify how much an explanation speaks in favor of (vs against) a prediction (§3.1, §3.2). On two datasets—TripAdvisor hotel review and Amazon product review spam detection (Ott et al., 2013; Hussain et al., 2020)—we find that 62%–86% of explanations from Llama-2 (Touvron et al., 2023), Mistral (Mistral, 2023), and Yi-1.5 (Young et al., 2024) are inconsistent, logically implying that they are also unfaithful (§6.1).

Second, we ask whether we can use our consistency measure PEX to train a model to produce more consistent—and therefore less unfaithful—explanations. We show how to apply both supervised fine-tuning and direct preference optimization (DPO) (Rafailov et al., 2024) to refine pre-trained LLMs (§3.4). For DPO, we sample explanations from LLMs and rank them according to PEX; those that score highly are treated as “preferred” in the optimization, and those that score lowly are treated as “dispreferred.” Experimentally, we show this DPO-based approach significantly improves on supervised fine-tuning and can improve explanation consistency by 43.1%–292.3% (§6.2).

It is, of course, possible that the fine-tuning leads to more consistent explanations, but not more faithful explanations. And so finally we measure whether the explanations that we optimized for consistency lead to improved faithfulness. We construct a proxy measure of faithfulness by assuming that a more faithful explanation should be more useful for an external observer to predict a model’s behavior (Lyu et al., 2024, see Section 3.4.2). Using this idea, we adopt a simulatability-based ex-

planation faithfulness evaluation method (Pruthi et al., 2022) (§4) and show that optimizing for PEX improves faithfulness by 1.5% to 9.7% (§6.3).

2 Related Work

Challenges in ensuring faithfulness of free-text explanations. Different from interpretable AI (Wallace et al., 2018; Selvaraju et al., 2017; Ribeiro et al., 2016; Briakou et al., 2023), generating explanations is a procedure to explicitly explaining model decisions to people (Miller, 2019). A faithful explanation should accurately reflect the reasoning process behind the model’s prediction (Jacovi and Goldberg, 2020; Lyu et al., 2024; Ribeiro et al., 2016). For models that first make a prediction with a standard black-box predictor and then justify the prediction with an explainer (Camburu et al., 2018; Park et al., 2018; Wu and Mooney, 2019), there is no guarantee for the explanation faithfulness. For models that first generate an explanation and then provided as the only input to the predictor, the explanations can still be self-inconsistent (Camburu et al., 2020; Zhou et al., 2023), or optimized in terms of plausibility instead of faithfulness (Kumar and Talukdar, 2020). Most methods that jointly explain and make prediction (Rajani et al., 2019; Narang et al., 2020; Ling et al., 2017; Jung et al., 2022; Ramnath et al., 2024) also do not ensure faithfulness, as models may ignore explanation during prediction. For chain-of-thought style prompting methods (Wei et al., 2022; Wang et al., 2022; Zhou et al., 2022a), the explanations can be unfaithful (Turpin et al., 2024; Lanham et al., 2023). Wiegreffe et al. (2022); Marasović et al. (2021) show potential for generating plausible free-text explanations with only a few examples, but the explanations can be unfaithful (Ye and Durrett, 2022; Chen et al., 2024).

Evaluate explanation faithfulness. We do not assume access to ground-truth explanations for evaluation, reflecting real-world scenarios; therefore white-box faithfulness evaluation (Zhou et al., 2022b; Chen et al., 2018) is not applicable to our problem. While robustness (Alvarez-Melis and Jaakkola, 2018) and perturbation-based evaluation methods (Samek et al., 2016) have been applied in the vision domain, their applicability to NLP is limited due to the discrete nature of language inputs, and removing a single word can render a sentence meaningless (Lyu et al., 2024). Simulatability-based methods have been employed

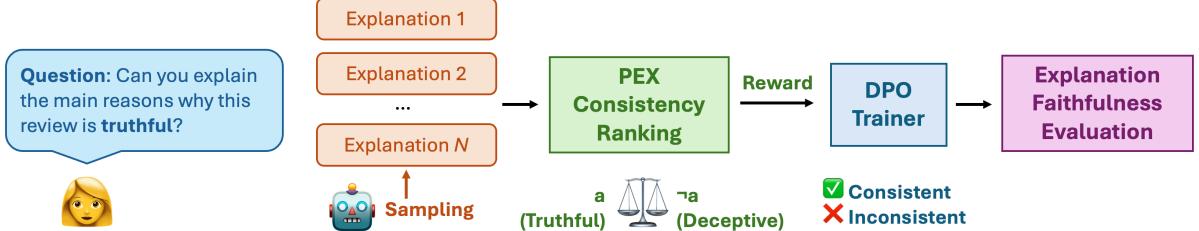


Figure 2: Our framework for generating more consistent explanations. Given a question and a model predicted answer, we sample explanations from a language model and rank them using Prediction-EXplanation consistency (PEX), which measures how well an explanation e supports a given prediction a compared to its negation $\neg a$. This measure serves as a reward signal to construct preference dataset for direct preference optimization (DPO), improving PEX consistency. Finally, we evaluate whether the consistency-optimized explanations are more faithful.

to evaluate explanation (Hase and Bansal, 2020; Hase et al., 2020; Doshi-Velez and Kim, 2017). This method is recommended to evaluate faithfulness (Lyu et al., 2024), as the more faithful an explanation is, the more information it should contain about the model’s decision mechanism, and thus the easier it would be for an external simulator, to predict the model’s behavior based on the explanation. The assumption is that if an explanation leads to a different prediction than that made by the model it explains, then it is unfaithful (Jacovi and Goldberg, 2020). Pruthi et al. (2022) propose a framework for evaluating explanations by measuring accuracy gains in a student model trained to mimic a teacher. This framework is designed for system-level evaluation rather than individual-level, as the student model requires more than one examples to learn from.

3 Measuring and Improving Prediction-Explanation Consistency

Faithful explanations that accurately reflect a model’s reasoning (Jacovi and Goldberg, 2020; Lyu et al., 2024) promotes AI transparency and accountability. In this work, we aim to improve the faithfulness of free-text explanations by optimizing their consistency. To this end, we first introduce PEX (Prediction-EXplanation consistency) measure to quantify how well an explanation speaks in favor of (vs against) a prediction (§ 3.2). We then use this measure as an optimization criterion to generate more consistent explanations (§ 3.4), as illustrated in Figure 2. Finally, we evaluate the efficacy of this optimization in generating more faithful explanations (§4).

3.1 Problem: Inconsistent Explanations

Our first goal is to develop a measure of whether a model’s explanations are *consistent* in justifying its predictions a over an its negation $\neg a$, as a *faithful* explanation cannot simultaneously support both predictions (Lipton, 1990; Miller, 2019; Brassard et al., 2024). We observe that language model generated explanations can be inconsistent, e.g. in Figure 1, the explanation “use a lot of positive language” supports both the truthful and deceptive hypotheses, failing to distinguish why the model predicted answer a instead of the alternative prediction $\neg a$. As a result, the explanation is not consistent and, therefore, not faithful. We focus on generating explanations to justify *opinion spam detection* (Ott et al., 2011; Hussain et al., 2020). We use this task because it requires minimal external knowledge while allowing for the generation of meaningful explanations.

We focus on language models that can answer questions, where a model M takes a question q as input and generates an answer a as output with probability $M(a | q)$. Additionally, we assume that upon request (e.g. prompting or other mechanisms) that the same model M can generate a potentially faithful explanation of its prediction. We denote this as $M(e | Q(q, a))$, where Q is an appropriate prompt, and explanation $e = (e_1, e_2, \dots, e_n)$ is a sequence of words. We use a one-shot example to prompt the model to generate explanation for its prediction. The explanation usually contains 2-3 rationales. The explanations in the one-shot example are generated by GPT-4 (detailed in §A.1).

For example, in opinion spam detection, question q is formatted as: $q = \text{Is this review truthful or deceptive? Review: } \{ \text{review} \}$. The output a is *Truthful* or *Deceptive*. If M predicts the answer a as *Truthful*, the prompt Q is formatted

as: $Q(\mathbf{q}, \text{Truthful}) = \text{Is this review truthful or deceptive? Review: \{review\}. Answer: Truthful.}$ Question: Can you explain the main reasons why the review is truthful? If M predicts the answer \mathbf{a} as **Deceptive**, the prompt Q is formatted as: $Q(\mathbf{q}, \text{Deceptive}) = \text{Is this review truthful or deceptive? Review: \{review\}. Answer: Deceptive.}$ Question: Can you explain the main reasons why the review is deceptive? An example of the generated explanation \mathbf{e} is **No specific examples: The review does not provide specific examples of the poor service or quality of the hotel.**

3.2 Measuring Prediction-Explanation (PEX) Consistency

We define explanation inconsistency as failing to explicate why the model predicted \mathbf{a} instead of its negation $\neg\mathbf{a}$. To quantify this, we introduce **Prediction-EXplanation** consistency (PEX), which measures how well an explanation supports a given prediction compared to its negation. We build on the *weight of evidence* framework (Melis et al., 2021; Good, 1985), extending it to compute probabilities when evidence is given as sequences of words. The weight of evidence measures the extent to which an explanation supports or undermines a hypothesis, making it a reasonable measure of explanation consistency.

PEX consistency. To measure the PEX consistency of explanation \mathbf{e} for a given question \mathbf{q} and model prediction \mathbf{a} , we compute a score to compare the likelihood of model M generating \mathbf{e} under different predictions, thereby quantifying how well the explanation supports the given prediction \mathbf{a} over its negation $\neg\mathbf{a}$:

$$C(\mathbf{e}) = \log \frac{M(\mathbf{e} | Q(\mathbf{q}, \mathbf{a}))}{M(\mathbf{e} | Q(\mathbf{q}, \neg\mathbf{a}))} \quad (1)$$

where the text prompt Q and explanation generation $M(\mathbf{e} | Q(\mathbf{q}, \mathbf{a}))$ are formatted in §3.1. We estimate the conditional probability $M(\mathbf{e} | Q(\mathbf{q}, \mathbf{a}))$ using the chain rule:

$$\begin{aligned} M(\mathbf{e} | Q(\mathbf{q}, \mathbf{a})) &= M(e_1 | Q(\mathbf{q}, \mathbf{a}))) \cdot \\ &\quad M(e_2 | Q(\mathbf{q}, \mathbf{a})), e_1) \cdot \\ &\quad \cdots M(e_n | Q(\mathbf{q}, \mathbf{a})), e_1, \dots, e_{n-1}) \end{aligned} \quad (2)$$

where e_i is the i -th word of the explanation \mathbf{e} . We estimate the conditional probability $M(\mathbf{e} | Q(\mathbf{q}, \neg\mathbf{a}))$ using the same approach.

If we view $M(\mathbf{e} | Q(\mathbf{q}, \mathbf{a}))$ as $M(\mathbf{e} | \mathbf{q}, \mathbf{a})$ by dropping the text format Q , PEX consistency $C(\mathbf{e})$ can also be computed using Bayes' rule as:

$$\begin{aligned} \log \frac{M(\mathbf{e} | \mathbf{q}, \mathbf{a})}{M(\mathbf{e} | \mathbf{q}, \neg\mathbf{a})} \\ = \log \frac{M(\mathbf{a} | \mathbf{q}, \mathbf{e})}{M(\neg\mathbf{a} | \mathbf{q}, \mathbf{e})} - \log \frac{M(\mathbf{a} | \mathbf{q})}{M(\neg\mathbf{a} | \mathbf{q})} \end{aligned} \quad (3)$$

where $\log \frac{M(\mathbf{a} | \mathbf{q}, \mathbf{e})}{M(\neg\mathbf{a} | \mathbf{q}, \mathbf{e})}$ is the posterior log-odds ratio between prediction \mathbf{a} and $\neg\mathbf{a}$ conditioned on question \mathbf{q} and explanation \mathbf{e} . $\log \frac{M(\mathbf{a} | \mathbf{q})}{M(\neg\mathbf{a} | \mathbf{q})}$ is the prior log-odds ratio, where $M(\mathbf{a} | \mathbf{q})$ is defined in §3.1. To compute $M(\mathbf{a} | \mathbf{q}, \mathbf{e})$ using a language model M , we format the prompt Q' as: $Q'(\mathbf{q}, \mathbf{e}) = \text{Is this review truthful or deceptive? Review: \{review\}. Analysis: } \mathbf{e}.$ Then, we compute adjusted PEX consistency $C'(\mathbf{e})$ as:

$$C'(\mathbf{e}) = \log \frac{M(\mathbf{a} | Q'(\mathbf{q}, \mathbf{e}))}{M(\neg\mathbf{a} | Q'(\mathbf{q}, \mathbf{e}))} - \log \frac{M(\mathbf{a} | \mathbf{q})}{M(\neg\mathbf{a} | \mathbf{q})} \quad (4)$$

We use *adjusted PEX consistency* (Eq 4) in our experiments because computing sequence probability for the original PEX consistency $C(\mathbf{e})$ (Eq 2), requires density estimation, which is often less reliable than computing classification probabilities for $C'(\mathbf{e})$. Additionally, sequence probability is typically affected by sequence length.

Supervised fine-tuning. We assume access to language models that provide output probabilities e.g. $M(\mathbf{a} | Q'(\mathbf{q}, \mathbf{e}))$, to compute PEX consistency (Eq 4). Pretrained language models, including LLaMA-2 and Mistral, do not perform well on opinion spam detection, with prediction accuracy close to random guessing. Thus we fine-tune the models on the training dataset using supervised fine-tuning training (SFT) to have better estimation of $M(\mathbf{a} | \mathbf{q})$. This is achieved by using the maximum-likelihood estimation (MLE) objective:

$$\sum_{(\mathbf{q}, \mathbf{a}) \in D} \log M(\mathbf{a} | \mathbf{q}) \quad (5)$$

where the training dataset D consisting of question answer pairs in the form of (\mathbf{q}, \mathbf{a}) . We use the fine-tuned models to measure PEX consistency.

3.3 Multiple Sampling for Generating More Consistent Explanations

The PEX measure (§3.2) can then be applied to rank language model generated explanations. For

each question q and the answer a predicted by model M , we sample explanations e from M using random sampling from the probability distribution $M(e | Q(q, a))$. We compute PEX consistency score (Eq 4) for each explanation, and rank the explanations according to the score.

3.4 Optimizing Explanation Consistency with Direct Preference Optimization

As the language models can generate inconsistent explanations for their predictions, we train models M_O using the direct preference optimization (DPO) objective (Rafailov et al., 2024) to generate more consistent explanations.

DPO training. Given a question q and answer a , M_O is trained to increase the likelihood $M_O(e_w | Q(q, a))$ of generating consistent explanation e_w as *preferred completion*, and decrease the likelihood $M_O(e_l | Q(q, a))$ of generating inconsistent explanation e_l as *dispreferred completion*. We use the language model M (§3.2) as base reference model, and use the DPO objective $\mathcal{L}(M_O; M)$:

$$\mathbb{E}_{(q, a, e_w, e_l) \sim D_O} \left[\log \sigma \left(\beta \log \frac{M_O(e_l | Q(q, a))}{M(e_l | Q(q, a))} \right) - \beta \log \frac{M_O(e_w | Q(q, a))}{M(e_w | Q(q, a))} \right] \quad (6)$$

where the weights of M_O are initialized from the base reference model M . σ is the logistic function, and β is a parameter controlling the deviation from the base reference model M , set to 0.1.

Preference dataset. To construct preference dataset D_O for DPO training, we select e_w and e_l from the explanations generated by the reference language model M . For each question q and answer a predicted by model M , we sample and rank the explanations using PEX consistency score (Eq 4), as described in §3.3. We consider explanations in the top $p\%$ are consistent, and those in the bottom $p\%$ are inconsistent. For each question and the answer, we use the combinations of the consistent and inconsistent explanations as preferred and dispreferred completions (e_w, e_l) to train DPO.

4 Explanation Faithfulness Evaluation

To evaluate the efficacy of optimizing PEX consistency (§3) in generating more faithful explanations, we construct a proxy measure of faithfulness by

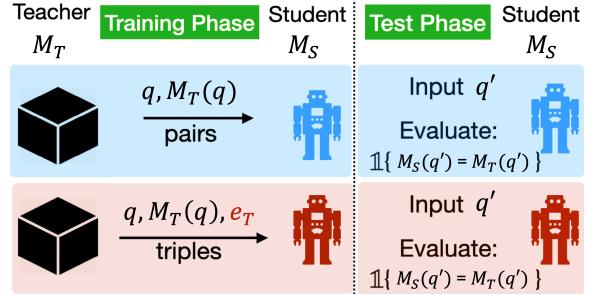


Figure 3: Explanation evaluation framework (figure reproduced from Pruthi et al., 2022): Student models learn from a teacher without explanation (top) or with explanation (bottom) as side information. Explanations e_T are effective if they improve student performance to mimic teacher model’s prediction on unseen examples during test phase without explanations (right).

assuming that a more faithful explanation should be more useful for an external observer to predict a model’s behavior (Lyu et al., 2024, see Section 3.4.2). Using this idea, we adopt a simulability-based explanation faithfulness evaluation method (Pruthi et al., 2022), as shown in Figure 3. In this framework, a teacher model generates a prediction and explanation, and a student model is trained to simulate the teacher’s model prediction. Explanations are available to the student model during training, but are not available during testing to avoid label leakage. We use language models as student models for our evaluation. While human evaluation is valuable, it is prohibitively expensive at scale, and may be confounded by humans’ preconceived notions—especially when conflating explanation faithfulness with plausibility. While language models also reflect biases, they offer a consistent and scalable means of evaluation.

Student model training. Our teacher model M_T is a supervised fine-tuned model M optimized for prediction accuracy (§3.2), and the student model M_S is initialized from a pretrained language model. We fine-tune the student model M_S to simulate the teacher’s model prediction and explanation: M_S takes question q as input, and generates answer a and explanation e autoregressively as its response, formatted as: $R(a, e) = \text{Answer: } a. \text{ Analysis: } e$. The model is trained to estimate $M_S(R(a, e) | q)$ using the MLE objective:

$$\sum_{(q, a_T, e_T) \in D_T} \sum_{n=1}^N \log M_S(R_n | R_{<n}, q) \quad (7)$$

where R_n is the n -th word of the text response $R(\mathbf{a}, \mathbf{e})$, and $R_{<n}$ is the first $n - 1$ words of the response. Dataset D_T consisting of samples in the form of $(\mathbf{q}, \mathbf{a}_T, \mathbf{e}_T)$, where $\mathbf{a}_T = M_T(\mathbf{q})$ is the answer predicted by the teacher model M_T , and explanation \mathbf{e}_T is generated by M_T to justify its prediction \mathbf{a}_T to the question \mathbf{q} . We generate the answer before the explanation for the student model, rather than after, because our preliminary experiments showed similar performance for both strategies. However, generating explanations at test time is computationally expensive. The answer-first strategy mitigates this overhead by enabling evaluation based on the likelihood of truthful or deceptive predictions without requiring explanation generation during inference.

Evaluation. During testing, the student model generates prediction $\mathbf{a}' = M_S(\mathbf{q}')$ for question \mathbf{q}' , and we measure the *simulation performance* by comparing its prediction \mathbf{a}' with the teacher model’s prediction $\mathbf{a}'_T = M_T(\mathbf{q}')$.

5 Experimental Setup

Datasets. We fine-tune pre-trained large language models to improve their prediction accuracy on two opinion spam detection datasets: (i) TripAdvisor hotel review dataset (Ott et al., 2013), which contains 800 truthful reviews and 800 deceptive reviews. We randomly split the dataset to obtain 960 pairs of (review, label) for training, 320 pairs for validation and 320 pairs for testing. (ii) Amazon product review dataset (Hussain et al., 2020), which we randomly select 1,000 truthful reviews and 1,000 deceptive reviews. We restrict reviews to those reviews containing at least 120 words to ensure that there is sufficient context for explanations. We split the selected reviews to obtain 1,200 pairs of (review, label) for training, 400 pairs for validation and 400 pairs for testing. We fine-tune the models on the train split, and select models on the validation split according to the F1 score.

We generate explanations using the fine-tuned models on the validation split to compute PEX consistency score statistics (§ 3.2). To evaluate explanation faithfulness, we also use the generated explanations on the validation split to serve as the teacher model to fine-tune a student model (§ 4). We report student model performance on the test split, evaluated against teacher predictions.

Pretrained-LLMs and SFT. We evaluate three different LLM architectures to generate explanations for the opinion spam detection datasets: (i) Mistral-7B-Instruct-v0.3 (Mistral, 2023), (ii) Llama-2-13B-chat (Touvron et al., 2023), and (iii) Yi-1.5-9B-chat (Young et al., 2024). We fine-tune the pretrained models (§3.2), achieving an F1 score of 94% on the TripAdvisor validation set and 93% on the Amazon validation set. Details for fine-tuning the models are given in Appendix §A.3.

DPO training. To construct the preference dataset for DPO training (§3.4), we generate 40 explanations per review-prediction pair in the training split (§3.3) using a sampling temperature of 1.0. To increase diversity, we sample 40 additional explanations at a temperature of 1.2. Explanations are ranked by their PEX consistency score (Eq 4), which measures how well they justify the prediction over its negation. For TripAdvisor dataset, we randomly sample 8 explanation pairs: preferred completions are drawn from the top 10% of scores and dispreferred from the bottom 10%, excluding pairs where both scores are above or below zero to ensure contrast. For Amazon dataset, we sample 8 pairs from the top and bottom 5% without applying a zero threshold, as many explanations for deceptive predictions have low scores. After this process, we obtain 3,565 training samples on the TripAdvisor dataset for DPO training with the Llama-2 model, 2,796 samples for the Mistral model, and 4,083 samples for the Yi-1.5 model. On the Amazon dataset, we obtain 9,600 training samples for each of the three language models. Additional details on model training are provided in §A.4.

Student model training. We train student models³ using k random examples from the validation split, where each example includes both the teacher model’s prediction and its generated explanation (§4). We perform a five-pass training for $k = 10$ and $k = 20$. We use small values of k because the student model can learn effectively from the prediction label alone when k is large. The student model shares the same architecture as the teacher model but is *not* fine-tuned on the training split. Further details are provided in §A.3.

Evaluation metrics. To measure how well the explanations justifies the prediction over its negation, we compute **PEX consistency** (Eq 4) on model

³ Our preliminary experiments do not suggest that using training-free in-context learning is fruitful.

generated explanations using the corresponding supervised fine-tuned model. To assess explanation faithfulness, we evaluate the trained student models using **simulation performance (F1 score)** on the test split, ensuring teacher model’s explanations are not provided as input to prevent label leakage (§4). The simulation F1 score is computed by using the teacher model’s predictions as ground-truth labels. We report average F1 score across all student model training passes.

6 Experiments

We investigate the following questions:

1. How consistent are the explanations generated by large language models (LLM)?
2. Can the consistency of LLM-generated explanations be improved?
3. Are explanations optimized for consistency also more faithful?

To address Q1, we analyze the PEX consistency score distribution of the explanations. For Q2, we train models using direct preference optimization to generate more consistent explanations. For Q3, we perform explanation faithfulness evaluation.

6.1 How consistent are the explanations generated by large language models?

Consistency distribution. Figure 4 shows the distribution of PEX consistency scores (§3.2) for explanations generated by the Mistral, Llama-2 and Yi-1.5 models on two opinion spam datasets. According to Good (1985); Melis et al. (2021), weight of evidence scores above 2 are considered consistent. For the Mistral model, we see that 85.6% of explanations are inconsistent (with PEX score < 0), 75.4% for the Llama-2 model, and 61.7% for the Yi-1.5 model. Since inconsistent explanations do not support the model’s predictions effectively, they pose a challenge for humans trying to understand how the model arrives at its decisions based on those explanations.

Sensitivity to prompt format. To assess whether the PEX consistency score is sensitive to different prompt formats, we use 3 different prompt formats (detailed in §A.2) to compute PEX consistency score and compute Kendall rank correlations between PEX consistency scores obtained from each pair of prompt versions. We find that PEX scores

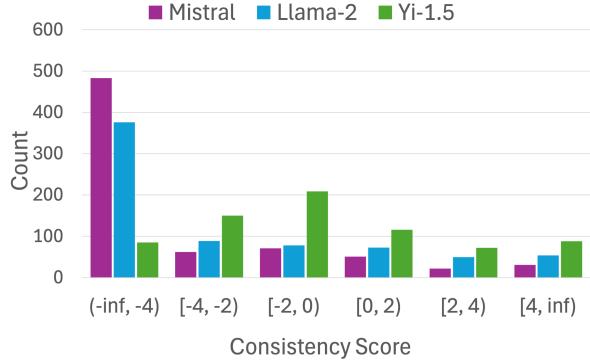


Figure 4: Prediction-EXplanation consistency (PEX) score distribution for different large language models. Explanations with PEX score higher than 2 are considered consistent (Good, 1985). **Takeaway:** models can generate 62%-86% inconsistent explanations.

Model	V1 vs V2	V1 vs V3	V2 vs V3
Llama-2	(0.42–0.52) [†]	(0.33–0.44) [†]	(0.52–0.60) [‡]
Mistral	(0.21–0.33) [†]	(0.48–0.56) [‡]	(0.38–0.48) [†]
Yi-1.5	(0.37–0.47) [†]	(0.50–0.58) [‡]	(0.40–0.50) [†]

Table 1: Kendall rank correlations of PEX consistency scores across prompt versions on the validation set. Ranges indicate 90% bootstrap confidence intervals. Correlations are moderate ([†]: mean ≥ 0.26) to strong ([‡]: mean ≥ 0.49). **Takeaway:** PEX scores are relatively robust to variations in prompt format.

are relatively robust to variations in prompt format, as shown in Table 1.

6.2 Can the consistency of LLM-generated explanations be improved?

We use direct preference optimization (DPO) to enhance explanation consistency (§3.4), the results are illustrated in Figure 5. For the Mistral model, DPO improves consistency by 2.8 to 5.1 points compared to explanations generated from the supervised fine-tuning (SFT) model (§3.2). DPO also improves 2.6 to 6.6 points for the Llama-2 model, and 1.8 to 3.8 points for the Yi-1.5 model. These results demonstrate the effectiveness of DPO in improving the PEX consistency of the generated explanations, by learning the patterns that contribute to more consistent explanations.

For the Mistral model on the TripAdvisor dataset, the average consistency scores of DPO-generated explanations remain below zero. This arises from the upper bound imposed by the sampled explanations being used to construct preference data: the highest consistency score among DPO-sampled explanations for this model averages -2.1 across the

Model	TripAdvisor			Amazon		
	Mistral	Llama-2	Yi-1.5	Mistral	Llama-2	Yi-1.5
Pred Only	55.5	59.6	63.0	83.2	88.6	67.8
+ SFT	66.2	58.3	66.0	85.2	89.2	75.5
+ DPO	69.8[†]	63.4[†]	70.9[†]	86.7[†]	91.6[†]	85.2[†]

Table 2: Simulation performance (F1) of student models on the TripAdvisor and Amazon test set, evaluating how well they approximate the teacher model’s predictions on unseen examples. The student models are trained using explanations from different teacher models: supervised fine-tuning (SFT) or direct preference optimization (DPO) with PEX consistency measure. [†] indicates results that are significantly higher than those of the SFT model, with $p < 0.05$ as determined by a two-related-sample t-test. §A.5 shows confidence intervals across training passes. **Takeaway:** optimizing PEX consistency improves explanation faithfulness.

reviews, which is lower than the Llama-2 model’s highest score of 4.4. Similarly, the highest consistency score among DPO-sampled explanations for Mistral model averages -2.8 on the Amazon dataset, which is lower than the Yi-1.5 model’s highest score of 0.3.

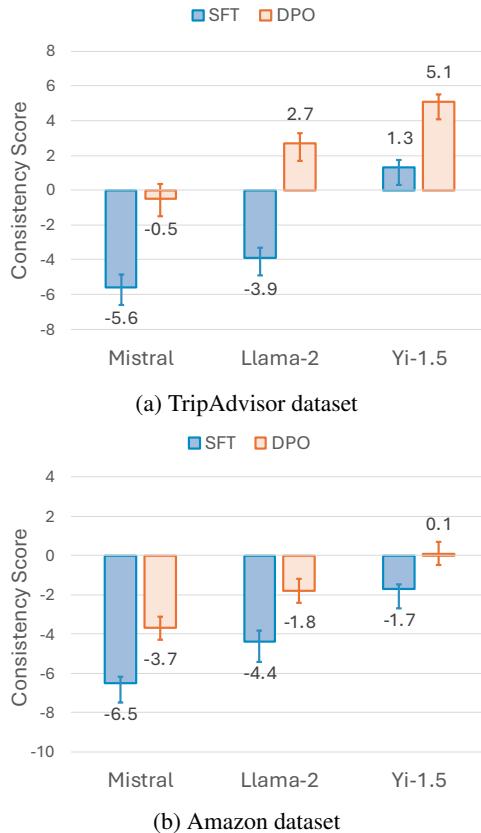


Figure 5: Explanation average PEX consistency scores for supervised fine-tuning (SFT) and direct preference optimization (DPO) models, computed on the TripAdvisor and Amazon validation datasets. The error bars represent 95% confidence intervals. DPO model scores are significantly higher than those of the SFT model ($p < 0.05$ from two-related-sample t-test). **Takeaway:** DPO training can improve explanation consistency.

6.3 Are explanations optimized for consistency also more faithful?

Systems. We evaluate the faithfulness of explanations (§4) generated by two systems: (i) *SFT*, a supervised fine-tuning model (§3.2), and (ii) *DPO*, a direct preference optimization model that improves PEX consistency (§3.4). Additionally, we report simulation performance for *Pred Only*, which provides only the teacher model’s prediction without any explanations during student model training. We use the same review across all three systems to enable a fair comparison.

Consistency-optimized explanations improve explanation faithfulness. Table 2 presents the student model’s simulation performance on the test set, where the student model was trained using explanations generated by different teacher models on the validation set. The F1 score measures how well the student model approximates the teacher model’s predictions on unseen data, averaged over different number of training samples (§5). On the TripAdvisor dataset, using DPO-generated explanations improves the F1 score by up to 14.3 points compared to not using explanations, and up to 5.1 points compared to using SFT explanations. The results indicate that explanations optimized using the PEX consistency measure with DPO better enable the student model to simulate the teacher’s predictions compared to explanations generated by the corresponding SFT model. On the Amazon dataset, using DPO-generated explanations improves the F1 score by up to 17.4 points compared to not using explanations, and up to 9.7 points compared to using SFT explanations. The DPO explanations from the Yi-1.5 model result in a larger F1 score gap compared to the SFT explanations, likely due to the model’s higher average PEX consistency score

Review: Booked through Hotwire and consequently was placed in a room where we barely had room to walk around our luggage. The 'view' was of the 'inner courtyard' of nothing but walls of windows of the other rooms. You were unable to determine what the weather was outside except that it was either day or night. They no longer have in-room coffee appliances... I would not stay there ever again, nor do I recommend it to anyone. I have had better stays at any Holiday Inn Express!

Question: Can you explain the main reasons why this review is truthful?

SFT output: [reason1] *Mentions an alternative hotel*: The reviewer suggests an alternative hotel (Holiday Inn Express) that they believe is a better choice, which is a practical and often truthful element in a hotel review. [reason2] *Provides specific details*: The reviewer mentions specific details about the room and its features, such as the 'inner courtyard' of nothing but walls, the in-room coffee appliances, the business hours of the concierge and center. (Consistency score=2.7)

DPO output: The review is truthful for the following reasons: [reason1] *Specific details provided*: The reviewer mentions the room size, the view, the in-room coffee, the internet access, the business center, and the Concierge services. [reason2] *Emotional response*: The reviewer clearly did not enjoy their stay and this is reflected in the language used, for example, using the phrase 'The 'gift' of nothing but wall of windows' suggests the author found this annoying... (Consistency score=2.0)

Figure 6: Direct preference optimization (DPO) generates better explanations: The Mistral SFT (supervised fine-tuning) model predicts the TripAdvisor review as truthful; however, its explanation includes the phrase *mentions an alternative hotel*, which is often associated with deceptive prediction (e.g. *Comparison to other hotels*: The reviewer *mentions other hotels that are "better" or have "better service"*), leading to inconsistency. In contrast, the DPO model produces more faithful explanations with a higher consistency score.

relative to the Mistral and Llama-2 models.

Qualitative examples. Figure 6 presents an example where the explanation generated by the DPO model receives a higher PEX consistency score than the explanation generated by the SFT model. The SFT-generated explanation is also less faithful, as it often supports the deceptive prediction. Additional examples are provided in §A.6.

7 Conclusion

We present a new consistency measure for free-text explanations as an important aspect of faithfulness. We optimize explanation consistency using direct preference optimization, and show that it can improve the consistency and faithfulness of model generated explanations. We hope this research advances explanation faithfulness and enhances transparency in AI-assisted decision-making. Another direction for future research is extending our approach to chain-of-thought explanations. Since our consistency measure and evaluation framework treat explanations and answers as independent variables, they are applicable regardless of how explanations are generated.

Limitations

Our approach to improving explanation consistency relies on computing the sequence output's conditional probability from language models. Consequently, this approach cannot be directly applied to explanations generated by completely black-box models. We evaluate explanation faithfulness at the system level rather than at the individual explanation level, as the faithfulness evaluation framework

we use is designed for system-level assessment. Our experiments focus on enhancing explanations for binary classification, but we envision future work extending this approach to multi-class classification, as weight of evidence can be applied in that scenario as well (Melis et al., 2021).

Acknowledgements

This material is based upon work supported by the NSF under Grant No. 2229885 (NSF Institute for Trustworthy AI in Law and Society, TRAILS), and U.S. Army Grant No. W911NF2120076. We thank Marine Carpuat, Navita Goyal, Amanda Liu, Connor Baumler, Ilaria Canavotto, John Horts, Eric Pacuit for discussion on explanation roles and evaluation. We also thank Nguyen X. Khanh for suggestion on direct preference optimization training, and Mingyang Xie for writing feedback.

References

David Alvarez-Melis and Tommi S Jaakkola. 2018. On the robustness of interpretability methods. *arXiv preprint arXiv:1806.08049*.

Ana Brassard, Benjamin Heinzerling, Keito Kudo, Keisuke Sakaguchi, and Kentaro Inui. 2024. *ACORN: Aspect-wise commonsense reasoning explanation evaluation*. In *First Conference on Language Modeling*.

Eleftheria Briakou, Navita Goyal, and Marine Carpuat. 2023. Explaining with contrastive phrasal highlighting: A case study in assisting humans to detect translation differences. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11220–11237.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. *Advances in Neural Information Processing Systems*, 31.

Oana-Maria Camburu, Brendan Shillingford, Pasquale Minervini, Thomas Lukasiewicz, and Phil Blunsom. 2020. **Make up your mind! adversarial generation of inconsistent natural language explanations.** In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4157–4165, Online. Association for Computational Linguistics.

Jianbo Chen, Le Song, Martin Wainwright, and Michael Jordan. 2018. Learning to explain: An information-theoretic perspective on model interpretation. In *International conference on machine learning*, pages 883–892. PMLR.

Yanda Chen, Chandan Singh, Xiaodong Liu, Simiao Zuo, Bin Yu, He He, and Jianfeng Gao. 2024. Towards consistent natural-language explanations via explanation-consistency finetuning. *arXiv preprint arXiv:2401.13986*.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

I. J. Good. 1985. Weight of evidence: A brief survey. *Bayesian statistics*, 2:249–270.

Tessa Han, Yasha Ektefaie, Maha Farhat, Marinka Zitnik, and Himabindu Lakkaraju. 2023. **Is ignorance bliss? the role of post hoc explanation faithfulness and alignment in model trust in laypeople and domain experts.** *Preprint*, arXiv:2312.05690.

Peter Hase and Mohit Bansal. 2020. **Evaluating explainable AI: Which algorithmic explanations help users predict model behavior?** In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5540–5552, Online. Association for Computational Linguistics.

Peter Hase, Shiyue Zhang, Harry Xie, and Mohit Bansal. 2020. **Leakage-adjusted simulatability: Can models generate non-trivial explanations of their behavior in natural language?** In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4351–4367, Online. 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. *arXiv preprint arXiv:2106.09685*.

Naveed Hussain, Hamid Turab Mirza, Ibrar Hussain, Faiza Iqbal, and Imran Memon. 2020. Spam review detection using the linguistic and spammer behavioral methods. *IEEE Access*, 8:53801–53816.

Alon Jacovi and Yoav Goldberg. 2020. **Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness?** In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online. Association for Computational Linguistics.

Jaehun Jung, Lianhui Qin, Sean Welleck, Faeze Brahman, Chandra Bhagavatula, Ronan Le Bras, and Yejin Choi. 2022. **Maieutic prompting: Logically consistent reasoning with recursive explanations.** In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1266–1279, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Maxime Guillaume Kayser, Bayar Menzat, Cornelius Emde, Bogdan Alexandru Bercean, Alex Novak, Abdalá Trinidad Espinosa Morgado, Bartłomiej Papiez, Susanne Gaube, Thomas Lukasiewicz, and Oana-Maria Camburu. 2024. **Fool me once? Contrasting textual and visual explanations in a clinical decision-support setting.** In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18891–18919, Miami, Florida, USA. Association for Computational Linguistics.

Sawan Kumar and Partha Talukdar. 2020. **NILE : Natural language inference with faithful natural language explanations.** In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8730–8742, Online. Association for Computational Linguistics.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilé Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. 2023. **Measuring faithfulness in chain-of-thought reasoning.** *Preprint*, arXiv:2307.13702.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. **Program induction by rationale generation: Learning to solve and explain algebraic word problems.** In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada. Association for Computational Linguistics.

Peter Lipton. 1990. Contrastive explanation. *Royal Institute of Philosophy Supplements*, 27:247–266.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions, nov. *NeurIPS*.

Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. 2024. Towards faithful model explanation in NLP: A survey. *Computational Linguistics*, pages 1–67.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. **Faithful chain-of-thought reasoning**. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 305–329, Nusa Dua, Bali. Association for Computational Linguistics.

Ana Marasović, Iz Beltagy, Doug Downey, and Matthew E. Peters. 2021. **Few-shot self-rationalization with natural language prompts**. In *NAACL-HLT*.

David Alvarez Melis, Harmanpreet Kaur, Hal Daumé III, Hanna Wallach, and Jennifer Wortman Vaughan. 2021. From human explanation to model interpretability: A framework based on weight of evidence. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 9, pages 35–47.

Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38.

Mistral. 2023. Mistral-7b-instruct-v0.3. <https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3>.

Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. 2020. Wt5?! training text-to-text models to explain their predictions. *arXiv preprint arXiv:2004.14546*.

Myle Ott, Claire Cardie, and Jeffrey T Hancock. 2013. Negative deceptive opinion spam. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: human language technologies*, pages 497–501.

Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T Hancock. 2011. Finding deceptive opinion spam by any stretch of the imagination. *arXiv preprint arXiv:1107.4557*.

Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Anna Rohrbach, Bernt Schiele, Trevor Darrell, and Marcus Rohrbach. 2018. Multimodal explanations: Justifying decisions and pointing to the evidence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8779–8788.

Danish Pruthi, Rachit Bansal, Bhuwan Dhingra, Livio Baldini Soares, Michael Collins, Zachary C Lipton, Graham Neubig, and William W Cohen. 2022. Evaluating explanations: How much do explanations from the teacher aid students? *Transactions of the Association for Computational Linguistics*, 10:359–375.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. **Explain yourself! leveraging language models for commonsense reasoning**. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy. Association for Computational Linguistics.

Sahana Ramnath, Brihi Joshi, Skyler Hallinan, Ximing Lu, Liunian Harold Li, Aaron Chan, Jack Hessel, Yejin Choi, and Xiang Ren. 2024. **Tailoring self-rationalizers with multi-reward distillation**. In *The Twelfth International Conference on Learning Representations*.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why should I trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. 2016. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626.

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

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. 2024. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36.

Eric Wallace, Shi Feng, and Jordan L. Boyd-Graber. 2018. **Interpreting neural networks with nearest neighbors**. In *BlackboxNLP@EMNLP*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Sarah Wiegreffe, Jack Hessel, Swabha Swayamdipta, Mark Riedl, and Yejin Choi. 2022. [Reframing human-AI collaboration for generating free-text explanations](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 632–658, Seattle, United States. Association for Computational Linguistics.

JF Woodward and L Ross. 2003. Scientific explanation: Stanford encyclopedia of philosophy. *Summer 2021 Edition*.

Jialin Wu and Raymond Mooney. 2019. [Faithful multimodal explanation for visual question answering](#). In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 103–112, Florence, Italy. Association for Computational Linguistics.

Xi Ye and Greg Durrett. 2022. The unreliability of explanations in few-shot prompting for textual reasoning. *Advances in neural information processing systems*, 35:30378–30392.

Alex Young, Bei Chen, Chao Li, Chengan Huang, Ge Zhang, Guanwei Zhang, Guoyin Wang, Heng Li, Jiangcheng Zhu, Jianqun Chen, et al. 2024. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, et al. 2022a. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

Yangqiaoyu Zhou, Yiming Zhang, and Chenhao Tan. 2023. [FLamE: Few-shot learning from natural language explanations](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6743–6763, Toronto, Canada. Association for Computational Linguistics.

Yilun Zhou, Serena Booth, Marco Tulio Ribeiro, and Julie Shah. 2022b. Do feature attribution methods correctly attribute features? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 9623–9633.

A Appendices

A.1 Explanation Prompting

We use the following GPT-4 generated explanation as a one-shot prompt to guide models in explaining the TripAdvisor hotel review dataset (see §3.1):

Review: We stayed at Hotel Monaco and upgraded to a suite. I was disappointed with the room layout, and shabby furniture. I bumped into the coffee table and the top almost crashed to the ground. The make up mirror in the bathroom didnt work, the TV reception was very poor and overall the room just wasnt worth the \$450 we spent there. Next time, we will go back to Trump Tower across the street, which is far superior to Hotel Monaco.

Explanation for truthful prediction: [reason1] Suggests an alternative hotel: The reviewer suggests an alternative hotel (Trump Tower) that they believe is superior, which is a practical and often truthful element in a hotel review. [reason2] Mentions specific details: The reviewer mentions specific details about the room, such as the room layout, shabby furniture.

Review: My stay at the Hotel Monaco recently was fantastic. We got a very reasonable price for our stay (especially since we were a AAA member), and there was plenty of perks to utilize; in-room spa treatments and an evening wine wine hour to name a few. Another thing that the kids really enjoyed was that, upon request, the staff brought us a fish in its own fishbowl for our room. How cool is that! The staff were all very helpful and friendly and our stay was very enjoyable. I would definitely recommend the Hotel Monaco to anyone in the downtown Chicago area.

Explanation for deceptive prediction: [reason1] Generic and overly positive tone: The review maintains a very positive tone without any specific details or occurrences that would substantiate such enthusiasm. The language used is quite general, lacking specific details or instances that typically accompany genuine experiences. [reason2] Uncommon or unrealistic details: The mention of receiving a fish in a fishbowl upon request is an uncommon and specific detail. While some hotels do offer unique services, the presentation of this detail, without further elaboration or context, seems exaggerated and implausible, casting doubt on the review's authenticity.

To guide models in generating explanations for the Amazon product review dataset (see §3.1), we use the following GPT-4 generated explanation as a one-shot example:

Review: I only wanted to use this to charge two phones from one adapter. This item at first seemed to be of decent quality. However, after 3 months of occasional use, the casing of one of the male plugs just fell apart in my hand as I was plugging it in. I did not apply undue force when it happened. After closer inspection, I could see how weak the plastic was - tiny, brittle plastic clips hold the case together. I might have been able to glue or tape it back together, but given that it charges \$500 phones, not worth the risk of a short. There is a diode and two wires soldered inside the housing that broke that could easily short circuit. I do not recommend this product if you value the items you are charging.

Explanation for truthful prediction: [reason1] Firsthand experience: The reviewer shares a personal account of using the product over three months, offering a credible, real-world perspective rather than speculation. [reason2] Safety concerns explained logically: The reviewer points out that the internal wiring and diode were exposed after the casing broke, increasing the risk of a short circuit.

Review: I was looking to downsize from an otterbox defender (whose holster's clip failed after one year) and this fits the bill nicely. Unlike the defender, the outside of the case is hard so it slides around (and in and out of your pocket) easily. You also don't have to open and close littler rubber doors to gain access to the headphone jack and charging/syncing port. The only hassle is that the rubber lining around the headphone jack wouldn't allow some of my older cheap cords or my good sony MDR 7506 headphones to plug in. Trimming the rubber solved the cheap cord problems but I had to replace the cord on the sonys with a hacked media bridge cord with a stepped plug on it.

Explanation for deceptive prediction: [reason1] Minimizes major flaws: The review glosses over a serious issue - the case blocks headphone and charging ports - by treating it as a minor fix. [reason2] Shifts blame: Suggests the problem is due to 'older cheap cords' rather than a design flaw, misleading readers.

A.2 Sensitivity to Prompt Format

To assess whether the PEX consistency score is sensitive to different prompt formats, we use 3 different prompt formats to compute the PEX consistency score described in Eq 4, where e is a free-text explanation to be measured:

V1: $Q'_1(q, e) =$ Is this review truthful or deceptive? Review: {review}. Answer: e .

Therefore, this review is

V2: $Q'_2(q, e) =$ Please classify the following review as either truthful or deceptive: {review}.

Answer: e . Hence, this review is

V3: $Q'_3(q, e) =$ Evaluate the authenticity of this review: {review}. Question: Is the review truthful or deceptive? Analysis: e . Answer: This review is

We then compute Kendall rank correlations be-

Model	Mistral			Llama-2			Yi-1.5		
	$k=10$	$k=20$	Avg	$k=10$	$k=20$	Avg	$k=10$	$k=20$	Avg
Pred Only	44.0 \pm 7.8	66.9 \pm 1.5	55.5	54.7 \pm 0.1	64.2 \pm 1.4	59.6	62.4 \pm 0.0	63.6 \pm 0.0	63.0
+ SFT	64.2 \pm 1.8	68.1 \pm 0.2	66.2	56.0 \pm 4.0	60.6 \pm 2.2	58.3	66.5 \pm 0.0	65.4 \pm 0.7	66.0
+ DPO	69.1\pm1.2	70.4\pm1.0	69.8	61.4\pm0.9	65.3\pm2.9	63.4	69.3\pm0.1	72.5\pm0.1	70.9

Table 3: Simulation performance (F1) of student models on the TripAdvisor test set, evaluating how well they approximate the teacher model’s predictions on unseen examples. The student models are trained using explanations from different teacher models: supervised fine-tuning (SFT) or direct preference optimization (DPO) with PEX consistency measure. The variable k represents the size of the teacher model’s training samples. The error bars after \pm represent 95% confidence intervals across 5 training passes.

tween PEX consistency scores obtained from each pair of prompt versions. The results are shown in Table 1, indicating PEX scores are relatively robust to variations in prompt format.

A.3 Supervised Fine-tuning and Student Model Training

For supervised fine-tuning (SFT), we finetune Llama-2-13b-chat (Touvron et al., 2023), Mistral-7B-Instruct-v0.3 (Mistral, 2023) and Yi-1.5-9B-chat (Young et al., 2024) models to improve their prediction accuracy on two opinion spam classification datasets (§3.2).

Hyperparameters and computation. For Llama-2, Mistral and Yi-1.5 models, we finetune our models for 20 epochs, using a learning rate of $2e^{-4}$, batch size of 2, and AdamW optimizer. We use a LoRA adapter (Hu et al., 2021) of rank 64 and alpha 16. We finetune the models using 1 RTXA6000 GPU, for ~ 24 hours. We train the student models for 100 epochs using 10 training examples and 50 epochs using 20 training examples, due to a small number of training samples. Training each model takes 1-2 hours.

Tools. We implement our models with Pytorch 2.0.1, Huggingface Transformers 4.31.0, scikit-learn 1.2.2 and SciPy 1.15.1.

Datasets. We two opinion spam detection datasets: (i) TripAdvisor hotel review dataset (Ott et al., 2013) and (ii) Amazon product review dataset (Hussain et al., 2020), under their Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License.

A.4 Direct Preference Optimization

Hyperparameters and computation. For Llama-2, we finetune our models using a learning

rate of $2e^{-4}$. For Mistral and Yi-1.5, we finetune our models using a learning rate of $2e^{-5}$. We finetune all the models for 5 epochs on the TripAdvisor dataset, and 1 epoch on the Amazon dataset. We use batch size of 2, AdamW optimizer, a LoRA adapter (Hu et al., 2021) of rank 32 and alpha 16. We finetune the models using 1 RTXA6000 GPU, for ~ 24 hours.

Tools. We implement DPO training using TRL (Transformer Reinforcement Learning) 0.14.0.

A.5 Explanation Faithfulness Evaluation

Table 3 and Table 4 show the simulation performance for explanation faithfulness evaluation, broken down by different values of k , which represents the number of training samples being used to train the student model.

A.6 Additional Qualitative Examples

Figure 7 shows additional qualitative examples for the explanations generated by DPO and SFT.

Model	Mistral			Llama-2			Yi-1.5		
	$k=10$	$k=20$	Avg	$k=10$	$k=20$	Avg	$k=10$	$k=20$	Avg
Pred Only	78.9±0.8	87.4±0.4	83.2	85.0±1.5	92.2±0.9	88.6	53.8±11.7	81.8±1.8	67.8
+ SFT	83.6±0.4	86.8±0.3	85.2	85.6±2.8	92.8±1.8	89.2	68.0±11.8	82.9±2.7	75.5
+ DPO	84.0±0.5	89.3±0.7	86.7	89.9±1.6	93.2±1.0	91.6	81.9±8.1	88.4±3.1	85.2

Table 4: Simulation performance (F1) of student models on the Amazon test set, evaluating how well they approximate the teacher model’s predictions on unseen examples. The student models are trained using explanations from different teacher models: supervised fine-tuning (SFT) or direct preference optimization (DPO) with PEX consistency measure. The variable k represents the size of the teacher model’s training samples. The error bars after \pm represent 95% confidence intervals across 5 training passes.

Review: Loved the case, it was good, absorbed impacts well, frosted back cover was nice and showed the battery doors design well. The camera hole and speaker hole had enough space, small hole for the microphone but it didn’t make a difference, feels comfortable when talking, headphone jack and micro-usb (yes you can slide it) are fully accessible. the point that the physical buttons were covered by the rubberized part made it a little tough to push them down but it provides security from dust going in there. feels a bit bigger but it slides in and out of the pockets just fine. provides good grip. covers ALL of the phones edge so dust and liquids are hard to get in there, doesn’t mean they won’t, after a while some dust may go in and you have to clean out the frosted area, but nothing tough.mines called it quits after about 9 months because the rubber started separating from the plastic, it arrived in an extremely great condition, didn’t feel cheap and the design was great down to the way the rubber and plastic are bonded. can be superglued together if it starts to rip (usually after a LONG time of extensive use). also has a nice hole incase if you put a small ornament or keychain thing on your phone. overall i recommend this. i got it from accessoriesgeeks but it’s the exact same product. love it. great build quality. real comfortable. totally recommended.

Question: Can you explain the main reasons why this review is deceptive?

SFT output: [reason1] Minimizes time of use: The review implies it broke after 9 months, but in reality it broke after a few days of use, which is a significant difference.\n[reason2] Overstates comfort: The review states it feels comfortable and has slides in and out of pockets, but in reality the physical buttons are too hard to press and the case does not slide in and out of pockets as easily due to the added bulk from the case.\n[reason3] Overstates durability: The review states it provides good protection and can be used for a long time, but in reality it cracks after a few days of use and the cracking is irreversible.\n[reason4] Uses too much superlative: The review uses too many positive adjectives (great, good, comfortable, slides in and out of pockets) without providing enough substance to back them up.\n[reason5] Lacks specifics: The review lacks specific details (e.g. how it broke, how long it was used, etc.) which makes it seem more credible, but in reality it is not. (Consistency score=-6.0)

DPO output: Sure! Here are three main reasons why the review is deceptive:\n1. Overstates the review period: The review claims to have owned the case for 9 months, but the language used (“arrived in an extremely great condition, didn’t feel cheap...”) and the praise heaped on the product’s build quality suggest that the reviewer is describing a much shorter period of ownership. This is likely an attempt to make the product appear better than it actually is by falsely claiming a longer period of ownership.\n2. Conceals the true condition of the product: The reviewer admits that the rubberized part started to separate from the plastic after 9 months of use, but claims that the problem can be solved by simply super-gluing it back together. This is misleading because it implies that the product is in good condition and can be easily fixed, rather than acknowledging that the item is actually damaged and should be replaced.\n3. Fails to provide a meaningful comparison: The reviewer claims that the product is “totally recommended” and that it has “great build quality,” but fails to provide any meaningful comparison to other cases on the market. The lack of a meaningful comparison makes the claim of “great build quality” seem artificially inflated, and the statement that the product is “totally recommended” is meaningless without some context. (Consistency score=-0.6)

Figure 7: DPO (Direct Preference Optimization) improves consistency score from the Llama-2 SFT (Supervised Fine-Tuning) model, but the generated explanation is still not consistent enough for its deceptive prediction on an Amazon product review.