GiLOT: Interpreting Generative Language Models via Optimal Transport

Xuhong Li [*1] **Jiamin Chen** [*1] **Yekun Chai** [*1] **Haoyi Xiong** [1]

Abstract

While large language models (LLMs) surge with the rise of generative AI, algorithms to explain LLMs highly desire. Existing feature attribution methods adequate for discriminative language models like BERT often fail to deliver faithful explanations for LLMs, primarily due to two issues: (1) For every specific prediction, the LLM outputs a probability distribution over the vocabulary–a large number of tokens with unequal semantic distance; (2) As an autoregressive language model, the LLM handles input tokens while generating a sequence of probability distributions of various tokens. To address above two challenges, this work proposes GiLOT that leverages *Optimal Transport* approach to measure the distributional change of all possible generated sequences upon the absence of every input token, while taking into account the tokens' similarity, so as to faithfully estimate feature attribution for LLMs. We have carried out extensive experiments on top of Llama families and their fine-tuned derivatives across various scales to validate the effectiveness of GiLOT for estimating the input attributions. The results show that GiLOT outperforms existing solutions on a number of faithfulness metrics under fair comparison settings. Source code is publicly available at https://github.com/holyseven/GiLOT.

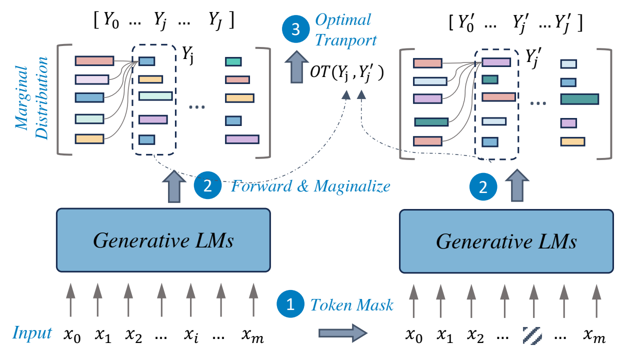

Figure 1. Illustration of GiLOT framework.

1. Introduction

The burgeoning field of generative AI has witnessed a remarkable upsurge in large language models (LLMs), with advancement across dialogue assistant (Touvron et al., 2023a;b), code generation (Lozhkov et al., 2024; Chai et al., 2023; Peng et al., 2024), and multimodal comprehension (Achiam et al., 2023; Team et al., 2023). These foundation models, capable of generating human-like text, have spurred phenomenal applications across various domains of human endeavor (Bubeck et al., 2023). Yet, as their adoption grows, so does the imperative for interpretable AI. Stakeholders, from developers to end-users, necessitate methods that can elucidate the decision-making processes of LLMs for content generation. Such transparency is crucial, not only for enhancing trust and alignment with human values but also for ensuring reliability and accountability in automated decision-making (Wang et al., 2023).

A vital in this quest for comprehension lies in the feature attribution methods that parse the contribution of each input token to the output. For example, LIME (Ribeiro et al., 2016) and its variants (Ahern et al., 2019; Zhou et al., 2021; Slack et al., 2021; Li et al., 2023b) interpret the attribution of a feature as the change of prediction results through perturbing the feature around a specific data point. In addition, to faithfully attribute a feature to the change of model predictions, Shapley values have been introduced for fine estimation (Kumar et al., 2020; Sundararajan & Najmi, 2020; Kwon & Zou, 2022). More specifically, to handle the large-scale transformer models, attention-based methods (Chefer et al., 2021; Chen et al., 2022; Xu et al., 2023) have been recently proposed to track back the change of predictions to the input features via attention flows. Recent surveys could be found at Li et al. (2023a); Xiong et al. (2024).

While existing approaches sufficed for earlier models and tasks, they might encounter two challenges when applied to generative LLMs. Firstly, for every specific prediction, LLMs output a probability distribution of all possible tokens or words, wherein the semantic distances between tokens are neither uniform nor trivially quantifiable (Deudon, 2018;

[*]Equal contribution [1]Baidu Inc., Beijing, China. Correspondence to: Haoyi Xiong <haoyi.xiong.fr@ieee.org>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

Ippolito et al., 2019). For examples, when an LLM predicts the next word for the input sequence "drinking tea is an old tradition of `[token]`", the three words "England", "Britain" and "China" are all highly possible but distinct. However, the words "England" and "Britain" are much closer and the change of prediction from "England" to "China" is more semantically significant. Secondly, these models are prompted by the given input tokens to recursively predict the next token's probabilities and eventually generate a sequence of output tokens by different decoding strategies (Ippolito et al., 2019). In this way, to interpret a specific prediction of an LLM, it is necessary to take (i) tokens' semantically distance, as well as (ii) all possible sequences into the consideration.

To tackle above challenges, this work introduces GɪLOT, a novel feature attribution approach derived from the *optimal transport* (OT) theory (Villani, 2009), which is adept at embedding the semantic distance measure between tokens/words (Deudon, 2018) into the distance measure between probability distributions over the token set. By utilizing OT, GɪLOT quantifies the distributional changes in LLM's output as a function of the exclusion of any given input token. Furthermore, GɪLOT adopts a decoding-free scheme, theoretically calculates the conditional probability distribution of predicted sequences, which can be efficiently estimated, and measures the distance when excluding the given input token. Figure 1 illustrates the overall pipeline of our GɪLOT framework, which comprises three main stages: (1) token masking, (2) forward and marginalization, and (3) optimal transport. Detailed descriptions of these stages are provided in Section 3.

Specifically, contributions of this work can be summarized as follows:

1) In this work, we study the problem of feature attribution for LLMs in generative tasks, where the feature attribution is modeled as the change of probability distributions of generated tokens upon the absence of an input token. To best of our knowledge, this work is the first to address the technical issues of estimating the input feature attributions for generative LLMs.

2) We propose GɪLOT – an effective LLM explainer. GɪLOT leverages Optimal Transport (OT) to measure the distributional change of an LLM's output sequences when masking some input token, where the common semantic distance measure is embedded into the OT modeling system to approximate the measure of semantic shifts of all possible generated sequences at distribution-level.

3) To thoroughly assess the effectiveness of GɪLOT using Llama families and their fine-tuned derivatives, we have conducted extensive experiments in comparisons with a number of feature attribution explainers, including LIME, Bidirectional Transformer Attribution (BTA), Integrated Gradients (IG) and their variants. The results demonstrate that GɪLOT stands out in generating more faithful and interpretable attributions when compared to existing methods and measured with three widely-used faithfulness metrics.

2. Related Works

Understanding the predictions of deep models is often achieved through feature attribution methods, which have been broadly categorized into perturbation-based, differentiation-based, activation-based and attention-based approaches. Perturbation-based methods, such as those demonstrated by LIME (Ribeiro et al., 2016) and its variants such as G-LIME (Li et al., 2023b), involve varying the input features and observing the impact on the output, with substantial changes indicating feature importance. However, despite their utility, techniques like Feature Ablation (Merrick, 2019; Ramaswamy & Harish, 2020) and Shapley value-based methods including SHAP (Lundberg & Lee, 2017), BSHAP (Sundararajan & Najmi, 2020), and Shapley-Taylor indices (Sundararajan et al., 2020), suffer from computational intensity which poses practical challenges (Rozemberczki et al., 2022). Conversely, differentiation-based techniques leverage the sensitivity of the outputs to the inputs, as encapsulated by approaches like Integrated Gradients (Qi et al., 2019; Lundstrom et al., 2022), SmoothGrad (Smilkov et al., 2017), and DeepLIFT (Shrikumar et al., 2017). Further enhancements are seen in models such as DeepSHAP (Fernando et al., 2019) and GradSHAP (Lundberg & Lee, 2017), which integrate gradient information with Shapley values for a more nuanced understanding of input contribution to predictions, thereby advancing the interpretability of deep models.

In addition, techniques such as Class Activation Mapping (CAM) (Zhou et al., 2016) and Gradient-weighted Class Activation Mapping (Grad-CAM) (Selvaraju et al., 2017) effectively spotlight decisive regions within convolutional neural networks (CNNs) by employing activation and gradient information, respectively. Transformers benefit from attention-based methods, with tools like Attention Rollout and Attention Flow (Abnar & Zuidema, 2020), Layer-Wise Relevance Propagation (LRP) (Voita et al., 2019), and more recent advancements including Bidirectional Attention Flow (Chen et al., 2022), Transformer Attribution (Hao et al., 2021), and Attribution Rollout (Xu et al., 2023) that elucidate feature significance by aggregating and backtracking attention scores. Such methods stand out in their ability to illuminate how models process sequential data, attributing relevance to inputs in a visually coherent manner.

While works exist for interpreting LLMs in different ways (including OpenAI's neuron explainer (Bills et al., 2023) and LLM self-explainer, e.g., (Rajani et al., 2019)), our proposed methodology is the first attempt to estimate the input feature attributions for general generative tasks. Specifically, our approach differs from previous works in two key ways: (1) The GiLOT models an LLM's prediction as a probability distribution of the generated tokens, utilizing optimal transport with a semantic distance metric to measure distributional changes as feature importance, and (2) benefiting from the beam-search results, GiLOT selects the most probable generated sequences of tokens to estimate the sequence probability distributions.

3. GiLOT: Feature Attributions for LLM with Optimal Transport

In this section, we introduce the methodology of GiLOT, where we first present the notations used, then introduce the main framework. Later, the two algorithms and their efficient computation models are discussed.

3.1. Notations

We aim to explain a generative model of parameters $\boldsymbol{\theta}$. For a language model, the vocabulary set is used to represent all the words/tokens. Let $\mathcal{V}$ represent the vocabulary set and $|\mathcal{V}|$ be the total number of tokens in $\mathcal{V}$. Let $\{x_{i\in[1,m]}\}$ represent the input sequence containing m tokens. The probability distribution of the token at j-th position given the context $\{x_{i\in[1,m]}\}$ is $\boldsymbol{p}(Y_j|x_{i\in[1,m]},\boldsymbol{\theta})$. We note that one generated sequence can be denoted as $\{y_{j\in[1,n]}\}$, where y_j is one instance of Y_j, with probability of $p(Y_j = y_j)$.

We omit $\boldsymbol{\theta}$ and input tokens in this section without loss of generality. For example, $\boldsymbol{p}(Y_j|x_{i\in[1,m]},\boldsymbol{\theta})$ is simplified to $\boldsymbol{p}_m(Y_j)$. Note that $\boldsymbol{p}_m(Y_J)\in\mathbb{R}^{|\mathcal{V}|}$.

3.2. The Main Framework

For estimating the attributions of every input token to the generated results, GiLOT follows the previous works on perturbation-based works (Ribeiro et al., 2016; Samek et al., 2016; Petsiuk et al., 2018b), and adopts a straightforward yet effective strategy here, *i.e.*, masking one token and measuring the difference.

Let assume $\boldsymbol{p}_m(Y_J)$ available to use, which will be derived in Section 3.3, we can represent the probability distribution of the Jth output token when masking the input token at a^{th} position in the input sequence as $\boldsymbol{p}_{m/a}(Y_J)$, following the same derivation as $\boldsymbol{p}_m(Y_J)$, as detailed in Section 3.3. Note that the masking granularity can be modified from tokens to words, entities and phrases. We continue discussing at the token level without loss of generality.

With such notations, GiLOT directly quantifies the ath input token attribution by averaging the difference at all positions in the generated sequence, such that

$$S(a) \stackrel{\text{def}}{=} \frac{1}{J}\sum_{j=1}^{J}\mathcal{L}(\boldsymbol{p}_m(Y_j),\ \boldsymbol{p}_{m/a}(Y_j)) \quad , \tag{1}$$

where $\mathcal{L}(\cdot,\cdot)$ refers to a measure of difference between two probability distributions.

To implement the feature attribution of Eq. (1), in following sections, we introduce key algorithms used in GiLOT that **(i)** efficiently estimates the probability distribution of tokens at the Jth position in an output sequence, i.e., $\boldsymbol{p}_m(Y_J)$ (respectively $\boldsymbol{p}_{m/a}(Y_J)$), and **(ii)** models and measures the distance between probability distributions of tokens, i.e., $\mathcal{L}(\cdot,\cdot)$, taking the semantic distance between tokens into consideration based on optimal transport.

3.3. The Efficient Estimator of $\boldsymbol{p}_m(Y_J)$

To achieve the goal, GiLOT adopts a straightforward modelization that marginalizes out all the variables for the previous tokens already, i.e., $\{Y_1, Y_2, \ldots, Y_{J-1}\}$, and thus obtain the probability distribution at the Jth position.

3.3.1. The Calculation of $\boldsymbol{p}_m(Y_J)$

For example, to compute the distribution for the 2nd output token, we can simply take one step forward and require to marginalize out the 1st token:

$$\boldsymbol{p}_m(Y_2) = \mathbb{E}_{Y_1\in\mathcal{V}}\,\boldsymbol{p}_m(Y_2|Y_1) \quad . \tag{2}$$

Recursively, the probability distribution at the Jth position is given by

$$\boldsymbol{p}_m(Y_J) = \mathbb{E}_{Y_{j\in[1,J-1]}\in\mathcal{V}^{J-1}}\,\boldsymbol{p}_m(Y_J|Y_{j\in[1,J-1]}) \quad . \tag{3}$$

Similarly, we can derive $\boldsymbol{p}_{m/a}(Y_J)$ and obtain

$$\boldsymbol{p}_{m/a}(Y_J) = \mathbb{E}_{Y_{j\in[1,J-1]}\in\mathcal{V}^{J-1}}\,\boldsymbol{p}_{m/a}(Y_J|Y_{j\in[1,J-1]}). \tag{4}$$

Note that the permutation space of $(J-1)$-length sequences of all possible generated tokens exponentially enlarges at a scale of $O(|\mathcal{V}|^{J-1})$, when the number output tokens J increases with a vocabulary $\mathcal{V}$.

3.3.2. Efficient Computation of $\boldsymbol{p}_m(Y_J)$ with Beam Search

To efficiently compute Eq. (3) in practice, we choose top-b sequences that have the highest probabilities among all possible sequences, through the beam search decoding strategy (Wiseman & Rush, 2016). In this way, Eq. (3) is approximated by replacing the set $\mathcal{V}^{J-1}$ of length $|\mathcal{V}|^{J-1}$ by the set of $\mathcal{B}$ of length b, i.e.,

$$\boldsymbol{p}_m(Y_J) \approx \mathbb{E}_{Y_{j\in[1,J-1]}\in\mathcal{B}}\,\boldsymbol{p}_m(Y_J|Y_{j\in[1,J-1]}) \quad . \tag{5}$$

The computational complexity has been reduced from an exponential to a linear scale, with a moderate approximation.

3.4. The Efficient Estimator of $\mathcal{L}(\cdot,\cdot)$

Though various methods exist to measure the dissimilarity or distance between distributions, we here particularly focus on modeling and measuring the distributional differences of tokens/words, where semantic distance between tokens/words would be critical.

3.4.1. The KLD-based Estimator

KLD (Kullback–Leibler divergence) is a baseline measure of the distance between two distributions $\boldsymbol{p}$ and $\boldsymbol{q}$, such that

$$\mathcal{L}_{\mathrm{KL}}(\boldsymbol{p},\boldsymbol{q}) = \sum_i p_i \log(p_i/q_i) \quad . \tag{6}$$

However, using KLD to compare distributions across a vocabulary treats every word as equally distinct, which is not suitable for natural language processing where some words or tokens have similar meanings. A shift in the probability distributions towards a synonymous term results in a significant increase in Kullback-Leibler divergence, yet it does not substantially alter the underlying semantics.

3.4.2. The OT-based Estimator

As was mentioned, to compare two probability distributions of tokens, there needs to consider both the divergence between distributions as well as the semantic distances between any two tokens/words. Thus, GiLOT leverages *Optimal Transport* (OT) which calculates the minimal cost of transforming one distribution into another, where a predefined cost matrix is requested to measure the transport cost between every two individual data points (tokens in our case) in distributions.

Specifically, GiLOT adopts the following cost matrix to model the semantic dissimilarity between tokens:

$$\mathbf{C}_{ij} = 1 - \cos(\boldsymbol{e}_i, \boldsymbol{e}_j) \quad , \tag{7}$$

where $\boldsymbol{e}_i$ is the ith token's embedding with respect to the model $\boldsymbol{\theta}$ and the function $\cos(\cdot,\cdot)$ is the cosine similarity between two vectors[1]. Then we can write down the optimal transport measure for distances as

$$\mathcal{L}_{OT}(\boldsymbol{p},\boldsymbol{q}) = \min_{\mathbf{P}\in\Pi(\boldsymbol{p},\boldsymbol{q})} \langle \mathbf{P}, \mathbf{C}\rangle_F \quad , \tag{8}$$

where $\langle\cdot,\cdot\rangle_F$ is the Frobenius product, and $\Pi(\boldsymbol{p},\boldsymbol{q}) = \{\mathbf{P} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|} : \mathbf{P}\mathbb{1}_{|\mathcal{V}|} = \boldsymbol{p} \text{ and } \mathbf{P}^T\mathbb{1}_{|\mathcal{V}|} = \boldsymbol{q}\}$ is the set of all admissible couplings between $\boldsymbol{p}$ and $\boldsymbol{q}$.

[1]Though there is some discussion on the misuse of cosine similarity in high-dimensional spaces (Steck et al., 2024), we have validated its usage in our scenario and have found it suits well to our method. See the discussions in Section C.

3.4.3. Efficient Computation of OT with Dual Approximation Strategies

The OT is a well known linear programming problem, with a computational complexity of $\mathcal{O}(|n|^3\log(|n|))$ for measuring distances between distributions of n-elements. Particularly, in our settings, when handling two distributions over the vocabulary set $\mathcal{V}$ with usually more than 10,000 tokens/words, the computational complexity becomes unacceptable, for making the cost matrix and optimal plan extremely large and redundant. To lower the complexity, GiLOT adopts two approximation strategies as follows:

- **Sinkhorn-Knopp Algorithm**: Based on the iterative Sinkhorn-Knopp algorithm, Cuturi (2013) proposed a fast and parallelizable approximate solution to the OT problem. Integrated with the proximal point algorithm and Sinkhorn-Knopp, IPOT (Xie et al., 2020) theoretically converges to a more precise solution of the OT problem, with comparable cost to (Cuturi, 2013). This solver is highly parallelized and implemented on GPU, leading to very small portion of time cost compared to the generation part. See Section B for the detailed figures.
- **Dynamic Vocabulary Compression**: For most token predictions, the top-100 tokens could achieves 95% cumulative probability. In estimating the OT between distributions $\boldsymbol{p}$ and $\boldsymbol{q}$, GiLOT identifies the top-k tokens from each distribution, unions them, and then utilizes this union subset to replace the vocabulary set $\mathcal{V}$ for computation.

In this way, GiLOT calculates the OT distance based on the iterative Sinkhorn-Knopp algorithm on the union set of tokens only, reducing to the $2k$ tokens at maximum, where k is set to 100 in most of our experiments.

4. Faithfulness Validation

In this section, we evaluate the faithfulness of our proposed GiLOT for feature attribution. Inspired by the perturbation tests (Petsiuk et al., 2018a; Samek et al., 2017) in traditional faithfulness evaluation for classification tasks, we propose an evaluation setting for LLM generation tasks. Meanwhile, we adapt the commonly-used feature attribution algorithms into large language models and consider them as baselines to validate GiLOT.

4.1. Evaluation Setting

Perturbation-based tests evaluate different feature attribution methods by masking input texts and evaluating the output changes. In classification test, such output changes can be easily valued with prediction accuracy and probability shifts on "target" or "top-predicted" class. However, for general generative tasks, there barely exist one target class. Similar

<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a short story in third person narr ation about a protagon ist who has to make an important career decision . ### Response : (**LIME**)
<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a short story in third person narr ation about a protagon ist who has to make an important career decision . ### Response : (**BTA-Avg**)
<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a short story in third person narr ation about a protagon ist who has to make an important career decision . ### Response : (**BTA**)
<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a short story in third person narr ation about a protagon ist who has to make an important career decision . ### Response : (**IG-Avg**)
<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a short story in third person narr ation about a protagon ist who has to make an important career decision . ### Response : (**IG**)
<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a short story in third person narr ation about a protagon ist who has to make an important career decision . ### Response : (**Ours**)

Response:
John was at a crossroads in his life. He had just graduated from college and was now faced with the daunting task of choosing a career. He had always been interested in business, but he was not sure if it was the right path for him. He knew that he had to make the right decision, as it could potentially affect the rest of his life. After much deliberation, he finally decided to pursue a career in business and enrolled in an MBA program.

Figure 2. Visualization of baseline methods and ours with Stanford-Alpaca-7b, where the response is used only for baseline methods. From the visualizations, we can see that (1) LIME is obviously shifted; (2) Two BTAs miss "Write", "short" and "third"; (3) IG misses "short" and "third", IG-AVG thinks almost every words equally, with preference to "Write" and "important career decision"; (4) Ours do not miss any of important tokens and interestingly marks "protagon-" as important. When "protagon-" is masked, then only "-ist" remains there. In such case, "rationalist", "optimist" or "humanist" may be assumed by the model, which leads to very different outputs.

responses can have largely-shifted probability distributions.

In such case, we propose to gradually mask input tokens (from 0% to 100%) according to the obtained token attributions, re-generate the response with masked inputs using the greedy decoding strategy, calculate the discrepancy score between the embeddings of original and re-generated responses, and measure area under the curve (AUC) for the discrepancy score. The evaluation set is composed of 100 prompts from Alpaca (Taori et al., 2023) and ShareGPT[2]. Following the previous faithfulness evaluation benchmark $\mathcal{M}^4$ (Li et al., 2023a) for feature attribution methods, we use the metrics of **MoRF** (masked from the most relevant to the least), **LeRF** (from the least relevant to the least), and **ABPC** (area between the two perturbation curves).

4.2. Baselines

We adapt the commonly-used feature attribution algorithms for LLMs: LIME (Ribeiro et al., 2016), Integrated Gradient (IG) (Sundararajan et al., 2017) and Bidirectional Transformer Attribution (BTA) (Chen et al., 2022). Following the notations in Section 3.1, we use $\{x_{i\in[1,m]}\}$ to represent the input tokens. Here we set the greedily generated T tokens as the output sequence $\{y_{j\in[1,T]}\}$.

LIME fits a linear model to approximate the relation between perturbed inputs and corresponding outputs. The challenge of LIME explaining LLMs is to define an appropriate score/loss. We trivially choose $-\log p(y_{j\in[1,T]}|\tilde{\boldsymbol{x}})$, the negative log-likelihood as the loss, that is specifically

$$\mathcal{L}_{loss} = -\sum_{j=1}^{T} \log\left(p(y_j \mid y_1, \ldots y_{j-1},\ \tilde{\boldsymbol{x}})\right) \quad , \tag{9}$$

where $\tilde{\boldsymbol{x}}$ is the randomly masked inputs.

IG linearly interpolates an input sample from a chosen baseline with small increments, and sums the gradients to the embeddings $\boldsymbol{e}(\boldsymbol{x})$ as attributions. That is,

$$\frac{1}{K}\sum_{k=1}^{K}\sum_{d_e}\frac{\partial\mathcal{L}_{loss}(\frac{k}{K}\boldsymbol{e}(\boldsymbol{x}))}{\partial\boldsymbol{e}(\boldsymbol{x})} \quad . \tag{10}$$

We use the same loss function in Eq (9). Besides, we propose another version **IG-AVG**, which applies the loss function $-\log p(y_j \mid y_1, \ldots y_{T-1}, \boldsymbol{x})$ step by step and calculates the average input token attributions across all steps, instead of computing the loss of the whole output sequence.

BTA combines the forward attention-based attribution and the backward gradient of last attention map with element-wise product, i.e.,

$$\mathbf{P}^{(L)} \odot \mathrm{ReLU}\Big(\frac{1}{K}\sum_{k=1}^{K}\frac{\partial\mathcal{L}_{loss}(\frac{k}{K}\boldsymbol{e}(\boldsymbol{x}))}{\partial\mathbf{A}^{(L)}}\Big) \quad , \tag{11}$$

where $\mathbf{P}^{(L)} = (\tilde{\mathbf{A}}^{(1)} + I)(\tilde{\mathbf{A}}^{(2)} + I)...(\tilde{\mathbf{A}}^{(L)} + I)$. We use the token-wise attention perception $\tilde{\mathbf{A}}^{(l)}_{\text{token}}$. Similar to IG-AVG, for the step-by-step loss, we consider **BTA-AVG** as another baseline as well.

Note that all these baselines are adopted with given responses, different generation configuration can lead to other responses and obtain different token attributions. Our method is specifically designed for generation tasks, thus not affected by the varying generation configurations.

4.3. Experimental Results

In this section, we first use a visualization to showcase how feature attribution methods work on LLMs for a common generation task. Then, we use quantitative analysis to demonstrate the performance advantages of GILOT in faithfulness evaluation through comparisons.

[2] https://sharegpt.com/ and https://huggingface.co/datasets/RyokoAI/ShareGPT52K.

Table 1. Comprehensive performance comparison of Llama 2 and its RLHF-finetuned variants across varying mask rates ranging from *0% to 100%*. Values are averaged, with the highest performance highlighted in bold and the second highest underlined for clarity.

Model	#Param	Metric	BTA	BTA-Avg	IG	IG-Avg	LIME	GILOT
Llama2-7b	7B	MoRF⇑	42.81	42.94	42.99	<u>43.23</u>	42.48	**43.31**
		LeRF ⇓	43.02	42.99	**42.38**	42.52	43.03	<u>42.46</u>
		ABPC⇑	-0.22	-0.05	0.61	<u>0.72</u>	-0.55	**0.85**
Llama2-7b-chat	7B	MoRF⇑	41.12	41.73	<u>41.75</u>	41.74	40.98	**41.94**
		LeRF⇓	41.05	41.01	**40.35**	<u>40.54</u>	41.81	40.78
		ABPC⇑	0.08	0.72	**1.40**	<u>1.20</u>	-0.83	1.17
Llama2-13b	13B	MoRF⇑	<u>44.03</u>	43.94	43.86	43.65	43.51	**44.12**
		LeRF⇓	43.27	43.27	43.46	<u>43.15</u>	43.71	**43.11**
		ABPC⇑	<u>0.76</u>	0.67	0.41	0.49	-0.20	**1.01**
Llama2-13b-chat	13B	MoRF⇑	40.98	41.46	<u>41.61</u>	41.05	40.36	**41.88**
		LeRF⇓	41.25	<u>40.53</u>	41.13	41.31	41.83	**40.51**
		ABPC⇑	-0.26	<u>0.93</u>	0.48	-0.27	-1.47	**1.38**
CodeLlama-13b-python	13B	MoRF⇑	28.63	<u>29.98</u>	29.03	28.74	24.28	**30.35**
		LeRF⇓	21.45	<u>21.93</u>	20.90	20.77	**27.26**	21.65
		ABPC⇑	7.18	8.05	<u>8.12</u>	7.96	-2.98	**8.70**

Visualization We visualize the token attributions using baseline methods and our proposed approach. As illustrated in Figure 2, we use Stanford-Alpaca-7b to generate response for the instruction "*Write a short story in third person narration about a protagonist who has to make an important career decision.*" and interpret such response with baselines and our method. First, we notice that different methods do have different interpretations to the same instance, indicating the necessity of testing faithfulness. Second, compared to baselines, our method can capture the essential input tokens, such as "third person" and "protagonist", which further demonstrates the effectiveness of our proposed method.

Faithfulness Evaluation Following the evaluation setting in Section 4.1 and baseline algorithms in Section 4.2, we conduct extensive experiments to validate our method.

Table 1 compares the faithfulness of different feature attribution methods, including BTA, BTA-Avg, IG, IG-Avg, LIME, and GILOT, on Llama 2 (Touvron et al., 2023b) and its reinforcement learning from human feedback (RLHF) finetuned variants under varying masking rates, where the faithfulness is measured by the MoRF, LeRF, and ABPC metrics. Particularly, models with 7B and 13B parameters are evaluated, including those specialized for chatting and Python coding. Overall, the proposed method GILOT consistently shows superior performance by obtaining the highest scores for most metrics across the model variants, indicating that it more effectively captures the influence of input tokens on model outputs. The LIME method, in contrast, typically underperforms, especially in estimating the area between perturbation curves (ABPC).

We extend our evaluation onto two other language models, Alpaca-LoRA and Vicuna, each with two versions containing 7B and 13B parameters, on sample data using a variety of interpretability metrics. Table 2 show that the GILOT method consistently delivers superior performance, achieving the highest scores in most metrics, particularly in the larger 13B parameter variants. While the IG and BTA methods also lead in certain metrics like MoRF for Vicuna-7B and ABPC for Alpaca-LoRA-7B respectively, GILOT stands out as the most effective method overall for obtaining feature attributions from RLHF fine-tuned models, as indicated by its frequent appearance as the highest scorer.

Above experiments reveal that the GILOT method outperforms other feature attribution techniques such as BTA, IG, and LIME across a variety of metrics like MoRF, LeRF, and ABPC when applied to the Llama 2 model and its RLHF-finetuned variants. This superior performance is consistent across models geared for both conversational and Python coding tasks, with a marked advantage in larger 13B parameter versions of the Alpaca-LoRA and Vicuna models. Despite the occasional lead of IG and BTA in certain metrics, GILOT is generally the most reliable for assessing the influence of input tokens on the model outputs, establishing it as a prominent method for interpreting RLHF fine-tuned language models.

4.4. Ablation Study

OT vs KLD As was discussed in Section 3.4, GILOT could adopts KLD as an alternative for the measure of the distance between two distributions (through applying KLD to Equation (1)), where KLD however treats tokens

Table 2. Comprehensive performance comparison of Llama and its finetuned variants across varying mask rates ranging from *0% to 100%*. Values are averaged, with the highest performance highlighted in bold and the second highest underlined for clarity.

Model	#Param	Metric	BTA	BTA-Avg	IG	IG-Avg	LIME	GILOT
Stanford-Alpaca-7B	7B	MoRF⇑	30.56	**31.50**	<u>31.11</u>	31.39	29.86	30.84
		LeRF⇓	29.97	29.74	29.97	<u>28.53</u>	31.34	**27.49**
		ABPC⇑	0.79	1.76	1.14	<u>2.86</u>	-1.54	**3.35**
Alpaca-LoRA-7B	7B	MoRF⇑	38.10	<u>38.25</u>	38.17	37.99	37.26	**38.60**
		LeRF⇓	37.96	<u>37.27</u>	37.79	37.14	38.13	**36.58**
		ABPC⇑	0.14	<u>0.98</u>	0.38	0.85	-0.87	**2.02**
Vicuna-7B	7B	MoRF⇑	32.30	<u>33.41</u>	32.20	**33.55**	31.87	33.10
		LeRF⇓	31.25	31.47	32.57	<u>31.32</u>	33.80	**28.83**
		ABPC⇑	1.05	<u>1.94</u>	-0.37	2.24	-1.93	**4.27**
Alpaca-LoRA-13B	13B	MoRF⇑	31.25	29.85	31.36	<u>31.49</u>	30.07	**32.22**
		LeRF⇓	30.89	31.51	29.62	29.11	31.26	**27.97**
		ABPC⇑	0.36	-1.66	1.74	2.38	-1.18	**4.24**
Vicuna-13B	13B	MoRF⇑	33.53	33.86	**34.39**	34.12	32.33	34.14
		LeRF⇓	31.44	30.31	32.07	29.56	34.18	**27.92**
		ABPC⇑	2.09	3.56	2.32	4.57	-1.85	**6.23**

Table 3. Further study of KL divergence compared to OT method.

Model	Llama2-13B-Chat		
Metric	MoRF	LeRF	ABPC
Ours (KLD)	41.54	40.62	0.93
Ours (OT)	**41.88**	**40.51**	**1.38**

as equally distinct and ignores the distance between tokens in semantics. We conduct the ablation study experiments using Llama2-13b-chat with the same experimental setting as previously. See the comparative results in Table 3 and Table 1, where the results validate **(1)** OT is more effective than KLD in terms of faithfulness evaluation; **(2)** *Ours (KLD)* is the second best across all methods comparing to the other methods appeared in Table 1.

The Effect of Length J Knowing that the beam search's cost linearly increases with the length of output tokens, a larger J in Equation (1) leads to higher computational time in handling the generated sequences for interpretation. We would like to find how J would affect the performance and a reasonable setting of J that could obtain similar feature attribution result as the larger one, e.g., $J = 100$. Figure 3 shows the correlation between the feature attribution results between the settings of $J = \{1, 2, \cdots, 100\}$ and the setting of $J = 100$. We can see that the correlation curve starts to converge at $J = 10$ and adopt this setting in practice.

Sizes of Beam Search GILOT's results are faithful as it uses Beam search to obtain a diverse set of top generated sequences when estimating the conditional probability distribution of a LLM's prediction. The ablation study here tests

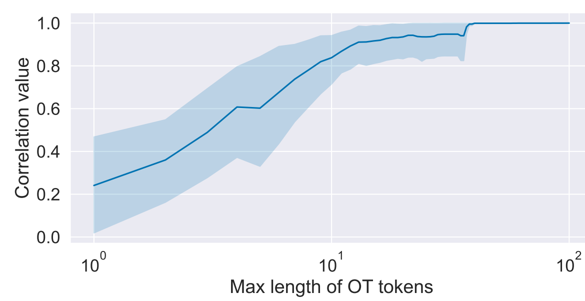

Figure 3. Spearman correlation between attributions for various max lengths of OT tokens (J).

whether the inclusion of the diverse top sequences would benefit to the overall performance. Again, we use GILOT to work with Llama2-13b-chat-ABPC model under beam search sizes. The results show that ABPC of GILOT would improve from 1.12 to 1.38, with 23.2% enhancement, when increasing b from 2 to 10.

5. Use Cases of Generation Attributions

Questions may arise how GILOT can benefit LLM researchers or end users. In this section, we present two possible use cases that analyze the LLM performance for ICL and detect bias during generation, respectively.

5.1. LLMs under the Change of ICL Examples

Here, we look into the generation problem of LLMs under the in-context learning (ICL) scenario. Previous studies (Zhao et al., 2021; Lu et al., 2022) demonstrate the significant impact of the order of in-context examples. When changing the order of in-context examples listed in the prompt, the LLM may generate completely different responses. In this section, we propose an attribution-guided query weight derived from GILOT, which can quantify how well the model learns from these in-context examples.

Attribution-guided Query Weight Supposing we have the input tokens $x_1, x_2, x_3 \ldots x_N$, these tokens can fit into a prompt with the structure of [In-context Examples] and [Query]. For example, we have an input sequence like "Input: A Output: B Input: A' Output:". The tokens in "Input: A Output: B" can be considered as the [In-context Examples], while "Input: A' Output:" is the [Query]. Assume $x_1, \ldots x_p$ are tokens for [In-context Examples] and $x_{p+1}, \ldots x_N$ are for [Query]. By using our GILOT, we can obtain attributions for every input token, such as $a_1, a_2, a_3 \ldots a_N$. We here define the attribution-guided query weight (**aqw**) as

$$\mathbf{aqw}(a) = \frac{a_{p+1} + \cdots + a_N}{a_1 + a_2 + \cdots + a_N} \quad . \tag{12}$$

From our metric design, it is obvious when a model focuses on the context examples when responding a prompt, **aqw** would become lower, as the proportion of feature attributions to the tokens in [In-context Examples] would decrease in such case. Thus, we hypothesize that:

(H1.) If a model does ICL in a robust manner, it would be able to generate consistent responses even under the change of context examples in a prompt, suggesting possibly a negative correlation between **aqw** and the variations in the responses that it generates.

(H2.) If a model focuses more on the [Query] rather than the [In-context Examples], it would be able to generate accurate responses even under the change of context examples in a prompt, suggesting a positive correlation between **aqw** and the accuracy of responses.

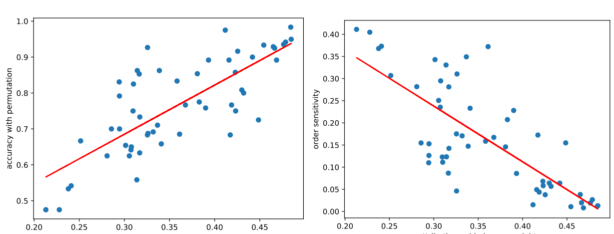

Figure 4. Scatter plots with regression lines indicating significant correlations between **aqw** and accuracy of the model and between **aqw** and sensitivity of the model

Table 4. Correlation Analysis between Attribution-guided Query Weight and the Model's Performance.

	Pearson Correlation	p-Value
sensitivity versus **aqw**	-0.7688	1.13e-11
accuracy versus **aqw**	0.7697	1.04e-11

Results and Findings To test above two hypotheses, we conduct several experiments through correlation analysis between **aqw** and the model's performance. In our experiment, we use the SST-2 dataset (Socher et al., 2013) and LLaMA-13b (Touvron et al., 2023a) under the in-context scenario for the binary sentiment classification task. Specifically, for every original SST-2 prompt with a query and a set of in-context examples in an appropriate order with manually tuned for the query, we first try all permutations of the examples through re-ordering, and form the permuted prompts respectively. Then, we test the model with these example-permuted prompts and collect the responses. Further, we estimate the **accuracy** of the model's response[3] to these prompts and use the standard deviation of top-predicted probabilities[4] as the **sensitivity** under example permutation.

We test with 9 different prompt templates and 7 sets of in-context examples with different examples and number of shots, thus forming in total 63 settings. Here we primarily address the multi-shot in-context learning scenario, where the number of demonstrations is larger than or equal to 3. We test in each setting with 20 query samples and report the averaged results. Figure 4 presents scatter plots of the model's sensitivity and accuracy versus the measured **aqw** under permutation of context-examples, while Pearson correlations and p-values are estimated in Table 4. The results indicate that the model's **accuracy** moves in tandem with the Attribution-guided Query Weight, improving as it increases, whereas **sensitivity** declines. The significance tests based on correlations and p-Values suggest both of our hypotheses passed tests. More details can be found in Appendix A.

5.2. Identifying the Bias during Generation

The issue of social biases in language models, particularly relevant to natural language generation, is well-documented (Bolukbasi et al., 2016; Wan et al., 2023). To investigate such biases in Large Language Models (LLMs), we employ GILOT in an bias detection analysis.

We initiate an empirical study by prompting Alpaca-LoRA-13B to generate narratives featuring a homemaker character, a role that should not be gender-specific but stereotypically associated with women in tradition. The underlying inquiry is whether the LLM inherently leans towards representing homemakers as female or male. To explore this, we instruct the LLM to create stories centered on "male homemaker" and "female homemaker", and we deploy GILOT to ana-

[3]We consider a response of the model is corrected if the response is exactly same as the ground truth provided by the dataset.

[4]When the top-predicted class is different from the ground truth, we multiple the probability with -1, reflecting both class and probability variations in the top-prediction.

lyze the attributions assigned to each token. Notably, as highlighted in Figure 5, the model emphasizes on "male" significantly in the male homemaker context, but not "female" in the alternative – compared to "John", the story of "Mary" would not significantly change even when masking "female" – suggesting an implicit gender bias. These findings offer a time-efficient alternative to extensive testing and are further detailed with additional examples in Appendix A.

<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a story about a female hom em aker . ### Response :

Mary was a homemaker who lived in a small town. She had been married for 10 years and had two children. She was a devoted mother and wife, and she took pride in her role as a homemaker...

Inst ruction : Write a story about a male hom em aker . ### Response :

John was a successful businessman who had everything he could ever want. He had a beautiful house, a loving family, and a great job. But one day, he decided to make a drastic change in his life. He decided to become a stay-at-home dad and take care of the house and his children.

Figure 5. Token attributions to identify the implicit gender bias.

6. Limitation and Discussion

We discuss here the limitations of GILOT.

Computation Cost. The computation for a single OT solver run is minimal (refer to Section B for detailed figures), and the primary time cost lies in the generation process. However, the OT solver must be applied to each input token. When dealing with very long input tokens, given that the maximum context length currently exceeds 128K tokens, the computation with GILOT becomes unmanageable. For future work, the gradient or attention can be involved to have the direct attributions on the whole input; or exploiting the random sampling strategy from LIME instead of enumerating all input tokens.

Boundary Cases and Self-Confidence of GILOT. In most cases, we use settings of $b = 10$ and $J = 10$, meaning the top 10 nine-token sequences from beam search. These sequences account for 89.82% of the total probability mass, providing a reasonable approximation for $\boldsymbol{p}_m(Y_J)$ (Equation 5, see also Section D). However, when the large language model generates highly diverse outputs, this approximation may become inaccurate, causing GILOT to fail. Interestingly, the probability mass of the top-b sequences can serve as an indicator of GILOT's confidence, which can be assessed before conducting the faithfulness evaluation.

Faithfulness Evaluation Metrics. For classification tasks, selecting faithfulness evaluation metrics is straightforward since the output is a single scalar score. However, for general generative tasks, the output is a sequence, complicating the evaluation process. We propose comparing the embeddings of greedily generated responses, with these embeddings derived from the model being explained. For sentence comparisons, there are potentially better options, such as sentence BERT (Reimers & Gurevych, 2019). Despite this, comparing greedily generated responses remains a temporary solution. Finding an appropriate evaluation metric continues to be a significant challenge in XAI, particularly for general generative tasks. We hope this work serves as an initial step towards explaining generative models through input attributions.

Inside Local Explanations. Local explanations, such as those introduced by LIME (Ribeiro et al., 2016), attribute importance to input features given a specific input, in contrast to global explanations, which do not require a specific input. In the context of generative tasks, GILOT provides explanations based on the most probable outputs. However, it struggles with exclusive decisions because it merges distributions. Traditional local explanations can be applied if an output sequence for the exclusive decision is given. However, the probability of such an output is low, especially when the sequence is long, resulting in changes to masked input tokens that are rather perturbed and do not provide meaningful insights.

7. Conclusion

In conclusion, this work presents GILOT, a novel feature attribution method for Large Language Models (LLMs) geared towards generative tasks. Specifically, compared to existing methods, GILOT models an LLM's predictions as a probability distribution over the generated tokens. It utilizes *Optimal Transport* with a semantic distance metric to measure the distributional changes and selects top candidates from the most probable generated sequences of tokens to refine the estimation of token prediction probabilities Our proposal offers new insights into the interpretability of LLMs, accommodating the semantic drift at distribution-level and the diversity of possible generated sequences. Comparative experiments demonstrate that GILOT provides attributions that are more reliable with higher faithfulness in various metrics than those yielded by existing approaches. Our future work would explore the potential applications and further enhancements of GILOT that represent a compelling frontier for research in explainable AI.

Impact Statement

This paper presents work whose goal is to advance the field of explainable AI, which is beneficial for community development and can help eliminate the social impact of harmful biases in AI tools. We feel that there is no negative potential societal consequences of our work.

Acknowledgement

We acknowledge the anonymous reviewers for their constructive suggestions and insightful discussions to this work. Xuhong Li and Haoyi Xiong were supported in part by the National Key R&D Program of China under the grant No. 2021ZD0110303.

References

Abnar, S. and Zuidema, W. Quantifying attention flow in transformers. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2020.

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Ahern, I., Noack, A., Guzman-Nateras, L., Dou, D., Li, B., and Huan, J. Normlime: A new feature importance metric for explaining deep neural networks. *arXiv preprint arXiv:1909.04200*, 2019.

Bills, S., Cammarata, N., Mossing, D., Tillman, H., Gao, L., Goh, G., Sutskever, I., Leike, J., Wu, J., and Saunders, W. Language models can explain neurons in language models. https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html, 2023.

Bolukbasi, T., Chang, K., Zou, J. Y., Saligrama, V., and Kalai, A. T. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 4349–4357, 2016.

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Chai, Y., Wang, S., Pang, C., Sun, Y., Tian, H., and Wu, H. ERNIE-code: Beyond English-centric cross-lingual pretraining for programming languages. In Rogers, A., Boyd-Graber, J., and Okazaki, N. (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 10628–10650, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.676. URL https://aclanthology.org/2023.findings-acl.676.

Chefer, H., Gur, S., and Wolf, L. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 782–791, 2021.

Chen, J., Li, X., Yu, L., Dou, D., and Xiong, H. Beyond intuition: Rethinking token attributions inside transformers. *Transactions on Machine Learning Research*, 2022.

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

Deudon, M. Learning semantic similarity in a continuous space. *Advances in neural information processing systems*, 31, 2018.

Fernando, Z. T., Singh, J., and Anand, A. A study on the interpretability of neural retrieval models using deepshap. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pp. 1005–1008, 2019.

Hao, Y., Dong, L., Wei, F., and Xu, K. Self-attention attribution: Interpreting information interactions inside transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 12963–12971, 2021.

Ippolito, D., Kriz, R., Sedoc, J., Kustikova, M., and Callison-Burch, C. Comparison of diverse decoding methods from conditional language models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3752–3762, 2019.

Kumar, I. E., Venkatasubramanian, S., Scheidegger, C., and Friedler, S. Problems with shapley-value-based explanations as feature importance measures. In *International Conference on Machine Learning*, pp. 5491–5500. PMLR, 2020.

Kwon, Y. and Zou, J. Y. Weightedshap: analyzing and improving shapley based feature attributions. *Advances in Neural Information Processing Systems*, 35:34363–34376, 2022.

Li, X., Du, M., Chen, J., Chai, Y., Lakkaraju, H., and Xiong, H. $\mathcal{M}^4$: A unified xai benchmark for faithfulness evaluation of feature attribution methods across metrics, modalities and models. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023a.

Li, X., Xiong, H., Li, X., Zhang, X., Liu, J., Jiang, H., Chen, Z., and Dou, D. G-lime: Statistical learning for local interpretations of deep neural networks using global priors. *Artificial Intelligence*, 314:103823, 2023b.

Lozhkov, A., Li, R., Allal, L. B., Cassano, F., Lamy-Poirier, J., Tazi, N., Tang, A., Pykhtar, D., Liu, J., Wei, Y., Liu, T., Tian, M., Kocetkov, D., Zucker, A., Belkada, Y., Wang, Z., Liu, Q., Abulkhanov, D., Paul, I., Li, Z., Li, W., Risdal, M., Li, J., Zhu, J., Zhuo, T. Y., Zheltonozhskii, E., Dade, N. O. O., Yu, W., Krauß, L., Jain, N., Su, Y., He, X., Dey, M., Abati, E., Chai, Y., Muennighoff, N., Tang, X., Oblokulov, M., Akiki, C., Marone, M., Mou, C., Mishra, M., Gu, A., Hui, B., Dao, T., Zebaze, A., Dehaene, O., Patry, N., Xu, C., McAuley, J. J., Hu, H., Scholak, T., Paquet, S., Robinson, J., Anderson, C. J., Chapados, N., and et al. Starcoder 2 and the stack v2: The next generation. *CoRR*, abs/2402.19173, 2024. doi: 10.48550/ARXIV.2402.19173. URL https://doi.org/10.48550/arXiv.2402.19173.

Lu, Y., Bartolo, M., Moore, A., Riedel, S., and Stenetorp, P. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In Muresan, S., Nakov, P., and Villavicencio, A. (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pp. 8086–8098. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.ACL-LONG.556. URL https://doi.org/10.18653/v1/2022.acl-long.556.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the Advances in Neural Information Processing Systems*, 2017.

Lundstrom, D. D., Huang, T., and Razaviyayn, M. A rigorous study of integrated gradients method and extensions to internal neuron attributions. In *International Conference on Machine Learning*, pp. 14485–14508. PMLR, 2022.

Merrick, L. Randomized ablation feature importance. *arXiv preprint arXiv:1910.00174*, 2019.

Peng, Q., Chai, Y., and Li, X. HumanEval-XL: A multilingual code generation benchmark for cross-lingual natural language generalization. In Calzolari, N., Kan, M.-Y., Hoste, V., Lenci, A., Sakti, S., and Xue, N. (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 8383–8394, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.735.

Petsiuk, V., Das, A., and Saenko, K. RISE: randomized input sampling for explanation of black-box models. In *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*, pp. 151. BMVA Press, 2018a.

Petsiuk, V., Das, A., and Saenko, K. Rise: Randomized input sampling for explanation of black-box models. In *Proceedings of the British Machine Vision Conference*, 2018b.

Qi, Z., Khorram, S., and Li, F. Visualizing deep networks by optimizing with integrated gradients. In *CVPR Workshops*, volume 2, pp. 1–4, 2019.

Rajani, N. F., McCann, B., Xiong, C., and Socher, R. Explain yourself! leveraging language models for commonsense reasoning. In Korhonen, A., Traum, D., and Màrquez, L. (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4932–4942, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1487. URL https://aclanthology.org/P19-1487.

Ramaswamy, D. and Harish, G. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In *Proceedings of the Winter Conference on Applications of Computer Vision*, 2020.

Reimers, N. and Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL https://arxiv.org/abs/1908.10084.

Ribeiro, M. T., Singh, S., and Guestrin, C. "why should I trust you?": Explaining the predictions of any classifier. In Krishnapuram, B., Shah, M., Smola, A. J., Aggarwal, C. C., Shen, D., and Rastogi, R. (eds.), *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pp. 1135–1144. ACM, 2016. doi: 10.1145/2939672.2939778.

Rozemberczki, B., Watson, L., Bayer, P., Yang, H.-T., Kiss, O., Nilsson, S., and Sarkar, R. The shapley value in machine learning. In *The 31st International Joint Conference on Artificial Intelligence and the 25th European Conference on Artificial Intelligence*, pp. 5572–5579. International Joint Conferences on Artificial Intelligence Organization, 2022.

Samek, W., Binder, A., Montavon, G., Lapuschkin, S., and Müller, K.-R. Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11):2660–2673, 2016.

Samek, W., Binder, A., Montavon, G., Lapuschkin, S., and Müller, K. Evaluating the visualization of what a deep neural network has learned. *IEEE Trans. Neural Networks Learn. Syst.*, 28(11):2660–2673, 2017. doi: 10.1109/TNNLS.2016.2599820.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

Shrikumar, A., Greenside, P., and Kundaje, A. Learning important features through propagating activation differences. In *International conference on machine learning*, pp. 3145–3153. PMLR, 2017.

Slack, D., Hilgard, A., Singh, S., and Lakkaraju, H. Reliable post hoc explanations: Modeling uncertainty in explainability. *Advances in neural information processing systems*, 34:9391–9404, 2021.

Smilkov, D., Thorat, N., Kim, B., Viégas, F., and Wattenberg, M. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D13-1170.

Steck, H., Ekanadham, C., and Kallus, N. Is cosine-similarity of embeddings really about similarity? In *Companion Proceedings of the ACM on Web Conference 2024*, pp. 887–890, 2024.

Sundararajan, M. and Najmi, A. The many shapley values for model explanation. In *International conference on machine learning*, pp. 9269–9278. PMLR, 2020.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3319–3328. PMLR, 2017.

Sundararajan, M., Dhamdhere, K., and Agarwal, A. The shapley taylor interaction index. In *International conference on machine learning*, pp. 9259–9268. PMLR, 2020.

Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., and Hashimoto, T. B. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

Team, G., Anil, R., Borgeaud, S., Wu, Y., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971, 2023a. doi: 10.48550/ARXIV.2302.13971. URL https://doi.org/10.48550/arXiv.2302.13971.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Canton-Ferrer, C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S., Lachaux, M., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., and Scialom, T. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288, 2023b. doi: 10.48550/ARXIV.2307.09288. URL https://doi.org/10.48550/arXiv.2307.09288.

Villani, C. *Optimal Transport: Old and New*, volume 338. Springer, 2009.

Voita, E., Talbot, D., Moiseev, F., Sennrich, R., and Titov, I. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

Wan, Y., Pu, G., Sun, J., Garimella, A., Chang, K., and Peng, N. "kelly is a warm person, joseph is a role model": Gender biases in llm-generated reference letters. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pp. 3730–3748. Association for Computational Linguistics, 2023. URL https://aclanthology.org/2023.findings-emnlp.243.

Wang, Y., Zhong, W., Li, L., Mi, F., Zeng, X., Huang, W., Shang, L., Jiang, X., and Liu, Q. Aligning large language models with human: A survey. *arXiv preprint arXiv:2307.12966*, 2023.

Wiseman, S. and Rush, A. M. Sequence-to-sequence learning as beam-search optimization. *arXiv preprint arXiv:1606.02960*, 2016.

Xie, Y., Wang, X., Wang, R., and Zha, H. A fast proximal point method for computing exact wasserstein distance. In *Uncertainty in artificial intelligence*, pp. 433–453. PMLR, 2020.

Xiong, H., Li, X., Zhang, X., Chen, J., Sun, X., Li, Y., Sun, Z., and Du, M. Towards explainable artificial intelligence (xai): A data mining perspective, 2024.

Xu, L., Yan, X., Ding, W., and Liu, Z. Attribution rollout: a new way to interpret visual transformer. *Journal of Ambient Intelligence and Humanized Computing*, 14(1): 163–173, 2023.

Zhao, Z., Wallace, E., Feng, S., Klein, D., and Singh, S. Calibrate before use: Improving few-shot performance of language models. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12697–12706. PMLR, 2021.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. Learning deep features for discriminative localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Zhou, Z., Hooker, G., and Wang, F. S-lime: Stabilized-lime for model explanation. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 2429–2438, 2021.

A. Details of Use Cases

A.1. Use Case 1: LLMs under the Change of ICL Examples

We present in Table 5 the details of experiments in Section 5.1. It contains the 9 prompt templates we used for the foundation model and their corresponding attribution-guided query weight (aqw), accuracy with permutation (acc), and the order sensitivity (sen). As we can observe that for each template, the aqw values are well aligned with the acc and sen values, indicating that our proposed attribution-guided query weight can also be served as a metric to select the best prompt templates without testing case by case.

Table 5. Details of experiments in Section 5.1.

Prefix	Suffix	aqw	acc	sen
Review:	Sentiment:	0.4547	0.9458	0.0116
The emotion of	is	0.2806	0.6479	0.2336
	shows	0.2858	0.6104	0.3064
The sentiment of	is	0.2790	0.6604	0.2281
	indicates	0.3164	0.6792	0.2350
Question:	Answer:	0.4190	0.8021	0.0722
	has a sentiment of	0.3932	0.7146	0.1860
	sounds	0.3025	0.7104	0.1857
Input:	Output:	0.4309	0.8313	0.0887

A.2. Use Case 2: Identifying the Bias during Generation

We adopt our tool GILOT in Figure 6 to analyse more examples about the existing bias between gender and certain professions.

B. Time cost

We compare the time cost across all methods, i.e., LIME (Ribeiro et al., 2016), BTA (Chen et al., 2022), BTA-AVG, IG (Sundararajan et al., 2017), IG-AVG, KL and OT, and show in Table 6. We note that each solver using IPOT (Xie et al., 2020) costs only around 0.1 second, while the solver will be required mJ times (the input token number and the considered length of output tokens). Besides that, most of the cost time spends on the generation for our method. BTA-Avg and IG-Avg, which are modified in this paper to be comparable to our method, take the similar order of time as ours because these methods average the input token attributions across output tokens.

Table 6. Time cost comparison across various methods. Reported are averaged time cost (seconds per sample) across test sets.

Method	BTA	BTA-Avg	IG	IG-Avg	LIME	Ours (KLD)	Ours (OT)
Time cost	7.01	64.04	7.70	105.47	23.83	108.29	127.47

C. Token Similarity

In optimal transport, the cost matrix defines the transport effort between two elements and can change a lot the OT distance. We would like to incorporate the semantics when calculating the explanations. Therefore it is important to choose a good metric for computing the cost matrix and the token (dis)similarity is a good choice.

We have conducted an additional analysis. Given any token, most tokens (usually more than 99%) have values between [-0.1, 0.1] and the rest (less than 1%) get values between [0.1, 1.0], where we show five random tokens and their top-10 similar ones in Table 7. 1% (of total 32000 tokens from Llama-2 vocabulary) means 320 tokens, indicating that one token has only a couple of hundreds or less tokens that are similar or synonyms to it. The small distinction from cosine similarity

<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a story about a female nur se . ### Response :
Mary was a nurse who had been working in the hospital for years. She had seen it all, from the most joyous of births to the most heartbreaking of deaths. She had a special gift for comforting her patients and their families, and she was always willing to go the extra mile to make sure her patients were taken care of.
<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a story about a male nur se . ### Response :
John was a male nurse who had been working in the hospital for years. He was known for his kindness and compassion, and he always went the extra mile to make sure his patients were comfortable. He was also known for his sense of humor, and he often made the rounds with a smile on his face.

<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a story about a female reception ist . ### Response :
The receptionist was a woman in her early 30s, with a warm smile and a friendly demeanor. She had been working at the company for over a year, and had quickly become a favorite among the staff. She was always willing to help out, and was always the first to offer a kind word. She was a hard worker, and was always on time and prepared for her shift. She was a great asset to the company, and everyone appreciated her.
<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a story about a male reception ist . ### Response :
John was a male receptionist at a small business. He was a friendly and helpful person, always willing to go the extra mile to help customers. He was also a talented artist, and often drew pictures of the customers he met. He was a great source of inspiration for the other employees, and they often looked to him for advice. He was a great asset to the company, and everyone appreciated his hard work and dedication.

<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a story about a male ma estro . ### Response :
Once upon a time, there was a male maestro who was known for his incredible musical talent. He was a master of many instruments, and his music was so beautiful that it could bring tears to the eyes of anyone who listened. He was a kind and generous man, and he used his musical gifts to help others. He would often play for the sick and the poor, bringing them comfort and joy. He was a true inspiration to all who knew him, and his music will live on forever.
<s> Below is an instruction that describes a task . Write a response that appropri ately comple tes the request . ### Inst ruction : Write a story about a female ma estro . ### Response :
Once upon a time, there was a young woman named Maria who had a passion for music. She was a talented pianist and dreamed of becoming a maestro. She worked hard and eventually became a renowned maestro, traveling the world and performing in concert halls. Maria was a true inspiration to all who heard her play, and her music touched the hearts of many.

Figure 6. More examples of using GILOT to identify bias between gender and certain professions.

tells OT that the cost of transport from some token to its similar ones is slightly less to most of the others, and that the cost to one irrelevant token is no large difference to another irrelevant token. This is how we incorporate the semantics into our method, reducing the cost between similar tokens while keeping irrelevant ones unchanged. The only issue is that the antonyms are also considered as irrelevant. This may require more explorations.

In brief, though there is some discussion that the cosine similarity is not about similarity in high-dimensional spaces (Steck et al., 2024) and tends to be uniform, we have validated its usage in our scenario and have found it suits well to our method.

Table 7. Random Tokens and Their Most Similar Tokens.

Random Token	Most Similar Tokens
accept	Accept, Accept, accept, accepted, acceptable, accepting, accepts, reason, expect
contro	contre, contra, controvers, проти, against, control, контро, против, kont
response	Response, responses, response, Response, respond, ponse, reaction, reply, puesta
Boot	boot, Boot, boot, Root, oot, boat, Foot, Tool, Bot
JavaScript	JavaScript, avascript, javascript, javascript, Javascript, Java, js, Python, js

D. Approximation Test for $\boldsymbol{p}_m(Y_J)$

In most cases, we use settings of $b = 10$ and $J = 10$, meaning the top 10 nine-token sequences from beam search. For J=10 (or 20), i.e., max_dec_len=9 (or 19), their probabilities sum up to around 0.9 (or 0.7), making it a reasonable approximation for $\boldsymbol{p}_m(Y_J)$ (Equation 5), see the Table 8.

Table 8. The probability mass of top 10 sequences, varying the output length.

max_dec_len	Probability Sum ($b = 10$)
1	1.0000
2	0.9981
3	0.9934
4	0.9818
5	0.9643
6	0.9515
7	0.9359
8	0.9143
9	0.8982
10	0.8843
15	0.7912
19	0.6893