Attention Meets Post-hoc Interpretability:
A Mathematical Perspective

Gianluigi Lopardo [1 2]   **Frederic Precioso** [1 3]   **Damien Garreau** [4]

Abstract

Attention-based architectures, in particular transformers, are at the heart of a technological revolution. Interestingly, in addition to helping obtain state-of-the-art results on a wide range of applications, the attention mechanism intrinsically provides meaningful insights on the internal behavior of the model. Can these insights be used as explanations? Debate rages on. In this paper, we mathematically study a simple attention-based architecture and pinpoint the differences between post-hoc and attention-based explanations. We show that they provide quite different results, and that, despite their limitations, post-hoc methods are capable of capturing more useful insights than merely examining the attention weights.

1. Introduction

The attention mechanism, introduced by Bahdanau et al. (2015), revolutionized neural networks by enabling models to focus dynamically on different parts of input sequences, enhancing their ability to capture long-range dependencies. This innovation laid the groundwork for various deep learning models. The Transformer architecture, introduced by Vaswani et al. (2017), is a notable application of the attention mechanism. Initially designed for natural language processing (NLP) tasks, the Transformer eliminated the need for recurrent neural networks and convolutional layers, relying solely on attention mechanisms. The Transformer has since become the state-of-the-art in numerous machine learning domains due to its flexibility, performance, and ability to model complex relationships in data. Its innovative design and significant improvements in training efficiency have paved the way for the development of advanced models such as BERT (Devlin et al., 2019), and GPT-3 (Brown

Figure 1. Different explainers can produce very different explanations. Here, the *attention* mean (α-**avg**) and maximum (α-**max**) over the heads, *LIME* (**lime**), the *gradient* mean (**G-avg**), L^1 norm (**G-l1**), and L^2 norm (**G-l2**), with respect to the tokens, and *Gradient times Input* (**G×I**) are employed to interpret the prediction of a sentiment-analysis model. Words with positive (respectively, negative) weights are highlighted in green (respectively, red), with intensity proportional to their weight. In the example, all the explainers identify the word *questionable* as highly significant, while only lime, and G×I highlight a negative contribution. Interestingly, α-avg and α-max identify the word *popular* as the most important word in absolute terms, in disagreement with the all others.

et al., 2020), which have revolutionized NLP.

As a by-product of the attention mechanism, weights corresponding to the per-token attention at a given layer can easily be extracted from the model. One is tempted to use these weights as explanations for the model's predictions, and many researchers have indeed done so (Chefer et al., 2021; Mylonas et al., 2023). However, the use of attention mechanisms for explainability has been questioned in the literature. Jain & Wallace (2019) notably critique its clarity, questioning the relationship between attention weights and model output. Conversely, Wiegreffe & Pinter (2019) argue that attention mechanisms remain useful for interpretability, without specifically addressing Jain & Wallace (2019)'s requirements. These works have sparked an intriguing debate in the literature, which we develop in Section 1.1. In our opinion, neither stance has provided a solid theoretical foundation to support their respective claims.

In this paper, we propose a mathematical analysis of an attention-based model and the associated explanations, trying to shed light on the respective merits of each approach

[1]Université Côte d'Azur, Inria, CNRS [2]LJAD [3]I3S [4]Julius-Maximilians-Universität Würzburg. Correspondence to: Gianluigi Lopardo <glopardo@unice.fr>.

not merely relying on experimental validation but truly looking at the connection between given explanations and the model's structure and parameters. Our analysis centers on a single-layer multi-head network, detailed in Section 2. This is a simplified variant of the transformer architecture proposed by Vaswani et al. (2017), tailored for a binary prediction task. Note that the binary classification restriction is illustrative and without loss of generality; the same results hold for multi-label predictions, when examining a specific class of interest. Additionally, while we focus on text classification tasks, we analyze token-level explanations, which could also be pixels in the context of Vision Transformers.

Specifically, we analyze the connections between attention-based and established post-hoc explanations. These methods include gradient-based, such as *Gradient* (Li et al., 2016), *Gradient×Input* (Denil et al., 2014), and perturbation-based approaches, such as LIME (Ribeiro et al., 2016).

We demonstrate that perturbation-based and gradient-based methods provide more insightful explanations than solely examining attention weights in Transformer models. We particularly concur with Bastings & Filippova (2020) that attention weights, while useful for input token weighting, can be misleading as model predictions' explanations, and advocate for post-hoc approaches.

Summary of the paper. In Section 1.1, we discuss the relevant literature, in particular focusing on the debate around attention-based explanations. We describe the model that we study in Section 2. In Section 3 we specifically discuss attention-based explanations. In Section 4 (resp. Section 5), we derive expressions for the gradient (resp. LIME) explanations associated to our model. These expressions (Theorems 4.1 and 5.1) are *explicit* with respect to the model parameters and the input document, thus allowing us to pinpoint exactly the differences between these approaches. In Section 6, we discuss the main limitations of our work, including the theoretical assumptions underlying the model under examination. We draw our conclusion in Section 7. All our theoretical claims are supported by mathematical proofs and empirical validation, detailed in the Appendix. The code for the model and the experiments are available at https://github.com/gianluigilopardo/attention_meets_xai.

1.1. Related work

The attention mechanism, pioneered by Bahdanau et al. (2015), enhanced neural networks' ability to focus on different parts of input sequences. Various forms of attention mechanisms exist, each characterized by distinct methods of query generation. A key differentiation lies in the computation of attention weights. Two methods are additive attention, as originally proposed by Bahdanau et al.

(2015), and **scaled dot-product attention** (on which we focus our study), introduced by Vaswani et al. (2017). Despite their differences, these two forms are theoretically similar (Vaswani et al., 2017) and have been found to yield comparable results (Jain & Wallace, 2019). This innovation paved the way for various deep learning models, including the Transformer architecture by Vaswani et al. (2017).

In essence, self-attention quantifies how much each token in a sequence is related with every other token. This relation is represented as attention weights, indicating the model's focus on different parts of the input. Thus, it is tempting to use these attention weights as explanations for the model's predictions. They provide a seemingly intuitive way to understand what the model is *paying attention to* when making a decision. Indeed, there are several methods to generate attention-based explanations. We delve into a discussion of these various methods in Section 3 of the paper. While attention weights offer valuable insights into the model's behavior, the use of attention mechanisms for explainability has been met with skepticism in the literature, generating an ongoing debate, that we summarize in the following.

The debate. A significant critique is offered by Jain & Wallace (2019), questioning the relationship between attention weights and model output. They argue, based on experiments across various NLP tasks, that attention weights do not provide meaningful explanations. In particular, Jain & Wallace (2019) proposes two properties that should hold if attention provides faithful explanations: (i) attention weights should correlate with feature importance measures (gradient-based measures and leave-one-out), and (ii) alternative (or counterfactual) attention weight configurations should yield corresponding changes in prediction. However, their experiments, suggest that these properties do not hold, leading them to conclude that attention weights are not suitable for interpretability.

On the other hand, Jain & Wallace (2019) has several limitations, first highlighted by Wiegreffe & Pinter (2019). Experimentally, Wiegreffe & Pinter (2019) conclude that "prior work does not disprove the usefulness of the attention mechanism for interpretability." They do not specifically address the claims presented by Jain & Wallace (2019), but they critique the experimental design proposed for point (ii), while somewhat agreeing with the first observation and corresponding experimental setup.

Specifically, Wiegreffe & Pinter (2019) introduce an end-to-end model training approach for finding adversarial attention weights. This approach ensures that the new, adversarial attention weights are plausible and consistent with the model. This is in contrast to the approach taken by Jain & Wallace (2019), where only the attention scores were changed, disrupting the model's training. Furthermore, Wiegreffe &

Pinter (2019) argue against the exclusive explanation that "attention is an explanation, not the explanation."

Serrano & Smith (2019) also scrutinize the use of attention for interpretability. They manipulate attention weights in pre-trained text classification models and analyze the impact on predictions. Their conclusion is that attention provides a noisy prediction of the input tokens' overall importance to a model, but it is not a reliable indicator.

More recently, Bibal et al. (2022) provide an overview of the debate on whether attention serves as an explanation, focusing on literature that builds on the works of Jain & Wallace (2019) and Wiegreffe & Pinter (2019). Bibal et al. (2022) argue that the applicability of attention as an explanation heavily depends on the specific NLP task. For instance, Clark et al. (2019) demonstrate that BERT's attention mechanism can provide reliable explanations for syntax-related tasks like part-of-speech tagging. Similar results are presented by Vig & Belinkov (2019) for GPT-2. In general, syntactic knowledge appears to be encoded across various attention heads and layers. Galassi et al. (2020) show that attention in transformers focuses on syntactic structures, making it suitable for global explanation.

Brunner et al. (2020) theoretically demonstrate that attention weights can be decomposed into two parts, with the *effective attention* part focusing on the effective input without being biased by its representation. This work is further expanded by Kobayashi et al. (2020) and Sun & Marasović (2021), who conduct a more in-depth evaluation. They find that alternative attention distributions obtained through adversarial training perform poorly, suggesting that the attention mechanism of RNNs indeed learns something useful. This finding contradicts the claim by Jain & Wallace (2019) that attention weights do not provide meaningful explanations.

It is important to note that there is currently no definitive theoretical support for either side of the debate on whether attention serves as an explanation. The positions presented by both Jain & Wallace (2019) and Wiegreffe & Pinter (2019) are primarily based on empirical experiments. The subsequent debate provided valuable insights and provoked thoughtful discussion, but did not conclusively prove or disprove the interpretability of attention mechanisms.

However, some recent works have investigated the role of attention through mathematical examination on specific tasks, in a similar fashion to our work. Wen et al. (2024) examines transformer interpretability by analyzing the model's weight matrices and attention patterns in the context of learning a *Dyck language* (Schützenberger, 1963). The authors demonstrate that vastly different solutions can be reached via standard training, cautioning against making interpretability claims based on inspecting individual components of the model. In particular, the attention pattern of a single layer

can be "nearly randomized" and still achieve high accuracy. In the same spirit, Li et al. (2023) provides a mechanistic understanding of how transformers learn semantic structure, through mathematical analysis and experiments on Wikipedia and LDA-generated (Blei et al., 2003) data. The study shows that both the embedding and self-attention layers can encode topical structures. In essence, even when the attention score is set to be uniform, the transformer can achieve a near optimal loss, as other parts of the model compensate for it. Finally, Cui et al. (2024) demonstrate that for a simple counting task (the *histogram task* defined in Weiss et al. (2021)), the loss landscape of a transformer with a dot-product attention layer and positional encodings reveals two distinct solutions: one with an attention matrix largely independent of the input tokens, and another that varies significantly based on the tokens and their semantic content. Ultimately, these works demonstrate that there is no evidence to assume that attention scores capture the core information underlying a transformer's predictions.

Post-hoc interpretability. Post-hoc interpretability refers to the process of explaining a model's predictions after it has been trained (Linardatos et al., 2021; Bodria et al., 2023). Among these techniques are gradient-based explanations (Li et al., 2016; Poerner et al., 2018; Arras et al., 2019; Atanasova et al., 2020; Denil et al., 2014; Sundararajan et al., 2017), that leverage the gradients of the model's output with respect to its input. Conversely, perturbation-based methods like LIME (Ribeiro et al., 2016), SHAP (Lundberg & Lee, 2017), Anchors (Ribeiro et al., 2018), or FRED (Lopardo et al., 2023b), examine the alterations in the model's output in response to changes in its input.

While post-hoc explanations offer valuable insights into machine learning models, they are not flawless. Their complexity can lead to inaccuracies, with sampling mechanisms potentially causing out-of-distribution issues and adversarial attacks (Hase et al., 2021; Slack et al., 2020). Some popular explainers have also been found lacking in soundness (Marques-Silva & Ignatiev, 2022). Thus, their use and interpretation necessitate careful scrutiny and continued research. Mardaoui & Garreau (2021) and Lopardo et al. (2023a) respectively propose deep theoretical analysis of LIME and Anchors for NLP models. In Section 5, we leverage Mardaoui & Garreau (2021)'s work to compute LIME coefficients for our model.

Attention meets post-hoc interpretability. Existing literature explores the differences between post-hoc and attention-based explanations, employing diverse methodologies and drawing varied conclusions. Ethayarajh & Jurafsky (2021) formally establish that attention weights are not Shapley values (Shapley, 1953), but attention flows (a post-processed version of attention weights) are, at least at

the layerwise level. Thorne et al. (2019) conduct a comparative analysis between post-hoc and attention-based methods. They select key features according to each explainer, subsequently using these features to make predictions and evaluate their accuracy. Their findings indicate that post-hoc methods like LIME and Anchors yield more accurate explanations than attention-based ones when implemented on an LSTM (Sak et al., 2014) for natural language inference.

Neely et al. (2021) evaluate the "agreement as evaluation" paradigm, comparing various explanation methods on Bi-LSTM (Huang et al., 2015) and Distil-BERT (Sanh et al., 2019) models. They conclude that consistency between different explainers should not be a criterion for evaluation unless a proper ground truth is available. This contradicts the agreement between Jain & Wallace (2019) and Wiegreffe & Pinter (2019). They also highlight theoretical limitations of state-of-the-art explainers and suggest using solid diagnostic tools like those proposed by Atanasova et al. (2020).

Neely et al. (2022) build on Neely et al. (2021)'s work, finding a lack of correlation among explanation methods, particularly in complex settings. They question the existence of an *ideal* explanation and the use of the *agreement as evaluation* paradigm for comparison, by demonstrating that similar explanations may not yield correlated rankings.

2. Attention-based classifier

In this section, we introduce the attention-based architecture that we study throughout the paper. We follow Phuong & Hutter (2022) in our presentation and notation. For any integer n, we set $[n] := \{1, \ldots, n\}$.

2.1. General description

In this paper, we consider a set of tokens belonging to a dictionary identified with $[D]$. A document ξ is an ordered sequence of tokens $\xi_1, \ldots, \xi_T$. We say that T is the length of the document. Without loss of generality, we can assume that the d unique tokens of ξ are the first d elements of $[D]$.

Our model f is a single-layer, multi-head, attention-based network, followed by a linear layer. More formally:

$$f(x) := \frac{1}{K} \sum_{i=1}^{K} f_i(x) = \frac{1}{K} \sum_{i=1}^{K} W_\ell^{(i)} \tilde{v}^{(i)}(x), \quad (1)$$

where $f_i := W_\ell^{(i)} \tilde{v}^{(i)} \in \mathbb{R}^{d_{\text{out}}}$, with $W_\ell^{(i)} \in \mathbb{R}^{1 \times d_{\text{out}}}$ the part of the final linear layer associated to head i, and for $i \in [K]$, $\tilde{v}^{(i)}(x)$ is the output of an individual head defined by Eq. (9). The value of f is used for classification; keeping in mind the sentiment analysis task described in the introduction, document ξ is classified as positive if $f(\xi) > 0$.

2.2. Attention

We now describe mathematically the self-attention mechanism at play in each head. Formally, we describe f_i for a given i, and thus temporarily drop the i index.

Token embedding. First, for each $t \in [T]$, the token $\xi_t = j$ is embedded as

$$e_t := (W_e)_{:,j} + W_p(t) \in \mathbb{R}^{d_e}, \quad (2)$$

where $W_e \in \mathbb{R}^{d_e \times D}$ is a matrix containing the embeddings of all tokens, where as $W_p : \mathbb{Z} \to \mathbb{R}^{d_e}$ is a deterministic mapping often called the *positional embedding*. It is common to set

$$\begin{cases} W_p(t)_{2i} & = \cos(t/T_{\max}^{2i/d_e}) \\ W_p(t)_{2i-1} & = \sin(t/T_{\max}^{2i/d_e}), \end{cases} \quad (3)$$

with $T_{\max}$ is the maximal document size. For all $T < t \leqslant T_{\max}$, the embedding of the fictitious token value is set to an arbitrary $h \in \mathbb{R}^{d_e}$, while the positional embedding remains the same. In other words,

$$\forall T < t \leqslant T_{\max}, \quad e_t := h + W_p(t) \in \mathbb{R}^{d_e}. \quad (4)$$

If $T > T_{\max}$, the last tokens are ignored and the input document is effectively discarded. We assume that the embedding matrices are *shared* between the K heads, but we want to emphasize that our analysis is easily amenable to different embedding matrices for each individual head.

Keys, queries, values. Next, these embeddings are mapped to *key*, *query*, and *values* vectors, defined respectively as

$$k_t := W_k e_t + b_k \in \mathbb{R}^{d_{\text{att}}}, \quad (5)$$

$$q_t := W_q e_t + b_q \in \mathbb{R}^{d_{\text{att}}}, \quad (6)$$

and

$$v_t := W_v e_t + b_v \in \mathbb{R}^{d_{\text{out}}}, \quad (7)$$

with $W_k, W_q \in \mathbb{R}^{d_{\text{att}} \times d_e}$, $W_v \in \mathbb{R}^{d_{\text{out}} \times d_e}$. For simplicity's sake, we will consider that the bias vectors $b_k, b_q \in \mathbb{R}^{d_{\text{att}}}$ and $b_v \in \mathbb{R}^{d_{\text{out}}}$ are all equal to zero.

Attention. For a given query $q \in \mathbb{R}^{d_{\text{att}}}$, each index t receives *attention*

$$\alpha_t := \frac{\exp\left(q^\top k_t / \sqrt{d_{\text{att}}}\right)}{\sum_{u=1}^{T_{\max}} \exp\left(q^\top k_u / \sqrt{d_{\text{att}}}\right)}. \quad (8)$$

We note that, since W_q and W_k are learnable parameters of the model, there is *a priori* no need for the $1/\sqrt{d_{\text{att}}}$ scaling factor. Nonetheless, it is instrumental in scaling the positional embedding properly and we keep it as is in our analysis.
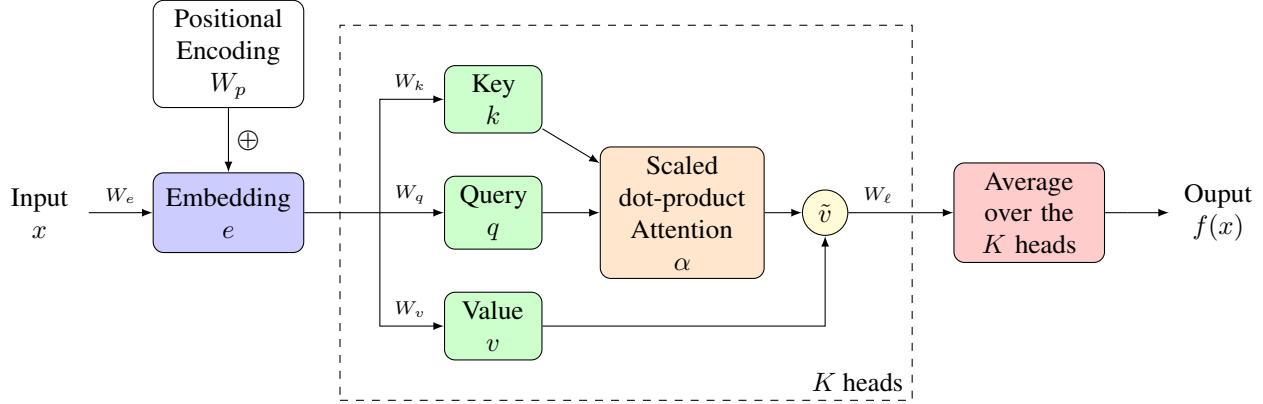
Figure 2. Illustration of the architecture of the model defined in Section 2. The input text, denoted as $x \in [D]^T$, is transformed into an embedding $e \in \mathbb{R}^{T \times d_e}$ by summing word embeddings and positional encodings as in Eq. (2). For each of the K heads, the key $k \in \mathbb{R}^{T \times d_{\text{att}}}$, query $q \in \mathbb{R}^{T \times d_{\text{att}}}$, and value $v \in \mathbb{R}^{T \times d_{\text{out}}}$ matrices are computed by applying linear transformations to e using $W_k, W_q \in \mathbb{R}^{d_{\text{att}} \times d_e}$, and $W_v \in \mathbb{R}^{d_{\text{out}} \times d_e}$, respectively. The attention weights $\alpha \in \mathbb{R}^T$ are then computed as the softmax of the scaled dot-product of k and q, as per Eq. (8). Then the intermediary output $\tilde{v} \in \mathbb{R}^{d_{\text{out}}}$ is computed are the average of the values v weighted by the attention α. Each head outputs the linear transformation $W_\ell \in \mathbb{R}^{1 \times d_{\text{out}}}$ of the $\tilde{v}$ associated with the query corresponding to the `[CLS]` token. The final prediction $f(x)$ of the model is the average of the outputs across all heads.

Output of the model. Finally, the intermediary output value before the final linear transformation associated to the query q is

$$\tilde{v} := \sum_{t=1}^{T_{\max}} \alpha_t v_t \in \mathbb{R}^{d_{\text{out}}} . \tag{9}$$

Each individual head is a linear transformation of the $\tilde{v}$ associated to the query corresponding to the `[CLS]` token (as in Devlin et al. (2019); Huang et al. (2015); Sanh et al. (2019)). Namely, as defined at the beginning of this section for $i \in [K]$, $f_i(x) = W_\ell^{(i)} \tilde{v}^{(i)}$.

We discuss limitations and the main differences between our model and practical architectures in Section 6.

3. Attention-based explanations

The **scaled dot-product attention**, introduced by Vaswani et al. (2017) (corresponding to the definition of Eq. (8)), essentially measures the relation among tokens. This results in the generation of a matrix, where each entry represents the degree of association between a pair of tokens. In essence, any attention-head $i \in [K]$ results in a $T \times T$ attention matrix $A^{(i)}$ (illustrated in Figure 3), where each entry $A_{s,t}^{(i)} = \alpha_t(q_s)$, *i.e.*, the attention as defined in Eq. (8) computed with respect to the s-th query token.

Furthermore, as this study focuses on a classification model, we only consider the `[CLS]` token, which encapsulates the core of the classification (Chefer et al., 2021). Formally, this implies that only the specific query q linked to the `[CLS]` token holds relevance in Eq. (8). This is equivalent to selecting the first row of the attention matrices in Figure 3.

Note that, in general, a Transformer model is structured as a series of sequential layers. Each of these layers is equipped with a specific number of parallel heads. These heads operate independently, executing the attention mechanism. Subsequently, in order to produce token-level attention-based explanations, one must aggregate the attention matrices at both the head level and layer level. Mylonas et al. (2023) offer a detailed depiction of the typical operations involved, and we specifically refer to Figure 2 in Mylonas et al. (2023) for a comprehensive picture.

In our scenario, the model defined in Section 2 is single-layered, hence we omit the layer-level aggregation.

As a result, for each head, there exists an attention vector of size T that emphasizes the focus of the head on each token. However, as depicted in Figure 3, heads often concentrate on different sections of the document. Therefore, the aggregation of the K attention vectors becomes a critical operation. The two most common operations at this level involve computing the average vector or determining the maximum value among the vectors for each token. Formally, we define, for any token $t \in [T]$,

$$\alpha\text{-avg}_t := \frac{1}{K} \sum_{i=1}^{K} \alpha_t^{(i)} , \tag{10}$$

and

$$\alpha\text{-max}_t := \max_{i \in [K]} \alpha_t^{(i)} . \tag{11}$$

Remark that, in general, α-avg and α-max can lead to very different explanations. Additionally, it is noteworthy that α-avg and α-max, G-l1 generate non-negative weights.
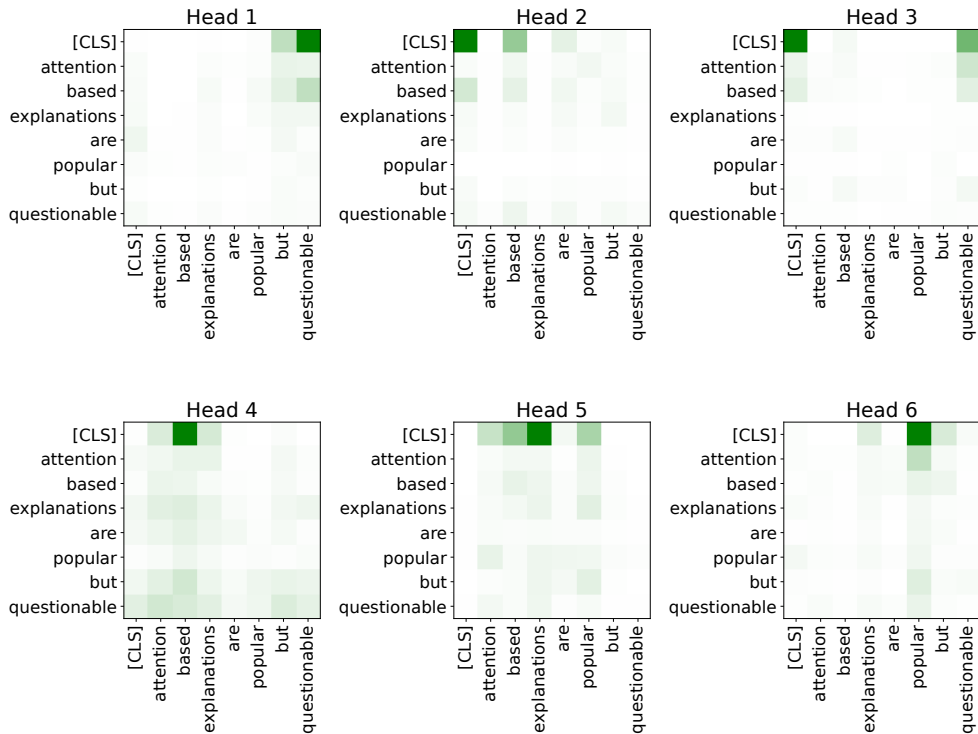
Figure 3. Attention matrices across the heads. Each head is represented by a distinct matrix, demonstrating the unique focus each head has on different parts of the document. The matrices illustrate that tokens within the document can carry significantly different weights, indicating the varying importance or relevance of each token in the context of the document. The aggregation of these weights to provide token-level scores is a critical aspect. Note that Eqs. (10) and (11) correspond to the average and the maximum values, respectively, of the first row across all six matrices.

Consequently, these methods do not differentiate between words that contribute positively or negatively to the prediction, as depicted in Figure 1.

4. Gradient-based explanations

In this section, we delve into the realm of gradient-based explanations. We first recap the main methods, before computing these explanations for the model proposed in Section 2.

4.1. Methods

In this section, we describe existing gradient-based methods, sometimes called saliency maps, by analogy to a similar technique in computer vision. Given a model f and an instance x, by a slight abuse of language, we call the gradient with respect to a token $t \in [T]$

$$\nabla_{e_t} f(x) \in \mathbb{R}^{d_e} . \tag{12}$$

It is important to note that the gradient ∇_{e_t} is calculated with respect to the embedding vector $e_t \in \mathbb{R}^{d_e}$. To derive per-token importance weights, several strategies exist. The primary approaches, falling into the class of *Gradient* explanations, involve taking the mean value (G-avg) (Atanasova

et al., 2020), the L^1 norm (G-l1) (Li et al., 2016), or the L^2 norm (G-l2) (Poerner et al., 2018; Arras et al., 2019; Atanasova et al., 2020) of the components of Eq. (12).

An alternative approach, known as *Gradient times Input* (G×I) (Denil et al., 2014), suggests computing salience weights by performing the dot product of the gradient from Eq. (12) with the input word embedding e_t. In our notation, the saliency weights are thus calculated as $e_t^\top (\nabla_{e_t} f(x))$.

While these methods share a common foundation, it is important to remark that the explanations they generate can vary significantly, and may even be contradictory. As illustrated in Figure 1, we observe, for instance, that G-l1 and G-l2 methods yield non-negative weights. In other words, these methods do not distinguish between words that contribute positively or negatively to the prediction, contrary to G×I (see Figure 1).

4.2. Gradient of our model

Let us consider the model described in Section 2. First, let us note that f is linear with respect to the f_i head, $i \in [K]$, hence, the gradient of f with respect to the token

embedding e_t is

$$\nabla_{e_t} f(x) := \frac{1}{K} \sum_{i=1}^{K} \nabla_{e_t} f_i(x) \in \mathbb{R}^{d_e} . \quad (13)$$

The quantity of interest is thus the gradient of a single attention-head, $\nabla f_i(x)$. Recall that q is the query corresponding to the classification token [CLS]. With this notation at hand, we can now state the following (which is proved in Appendix A.).

Theorem 4.1 (Gradient meets attention). *The gradient of the model f defined by Eq.* (1)*, with respect to the embedded token e_t, $t \in [T]$, is*

$$\nabla_{e_t} f(x) = \frac{1}{K} \sum_{i=1}^{K} \left[\alpha_t^{(i)} (W_v^{(i)})^\top (W_\ell^{(i)})^\top \right. \quad (14)$$

$$\left. + \frac{\alpha_t^{(i)}}{\sqrt{d_{att}}} W_\ell^{(i)} \left(v_t^{(i)} - \sum_{s=1}^{T_{max}} \alpha_s^{(i)} v_s^{(i)} \right) (W_k^{(i)})^\top q \right] \in \mathbb{R}^{d_e} .$$

By substituting Eq. (14) into Eq. (13), we are able to reconstruct the gradient-based explanations discussed in Section 4.1. Indeed, G-avg, G-l1, and G-l2, are nothing but the average, the L^1, and the L^2 norm of $\nabla_{e_t} f(x) \in \mathbb{R}^{d_e}$, respectively, while G×I is the dot product between the gradient and the embedding vector: $e_t^\top (\nabla_{e_t} f(x))$.

We now make a few comments. (i) the gradient of f, at the first order approximation, is **linear in α**, which can explain the correlation with the attention weights to some extent. Consequently, **α-avg is correlated with G-avg and G-l1**, as by desiderata (i) of Jain & Wallace (2019), while the same does not hold for α-max and G-l2. We want to emphasize that this may not necessarily hold true for deeper models. (ii) **the gradient captures the influence of the linear layers $W_\ell^{(i)}$**: we believe that this is a useful insight, disregarded by attention-based explanations. For instance, let us assume $K = 1$ and $v_t \approx \sum_{s=1}^{T_{max}} \alpha_s v_s$ (corresponding to a situation where the value vector is "typical"). Then the only remaining part in Eq. (14) is $\alpha_t W_v^\top W_\ell^\top$. A positive α_t can give rise to **negative** explanations if the coordinates of $W_v^\top W_\ell^\top$, reflecting the true behavior of the model.

5. Perturbation-based explanations: the example of LIME

Let us now turn towards perturbation-based explanations. We focus on LIME for text data, first recalling how it operates, introducing additional notation on the way, then stating our main result.

5.1. Reminder on LIME

Here we give a short introduction to LIME for text data in our context, following closely Mardaoui & Garreau (2021).

The overall idea underlying LIME for text data is to start from ξ the document to explain and produce local perturbations $X_1, \ldots, X_n$. From these perturbations, a local linear model trying to fit the predictions of f is trained, and the linear coefficients corresponding to this linear model given as explanation to the user.

Sampling. Let us call X the distribution of the randomly perturbed documents. Then, in our notation, X is generated as follows: first pick s uniformly at random in $[d]$ (the local dictionary), then chose a set $S \subseteq [d]$ of size s uniformly at random. Finally, remove from ξ all occurrences of words appearing in S. Here, removing means replacing by the UNK token. In the present work, for simplicity, we assume that tokens and words coincide. The perturbed samples $X_1, \ldots, X_n$ are i.i.d. repetitions of this process. Associated to the X_is we have vectors $Z_1, \ldots, Z_n \in \{0, 1\}^d$, marking the absence or presence of a word in X_i. Namely, we set $Z_{i,j} = 1$ if word j belongs to X_i and 0 otherwise.

Weights. Each new sample X_i receives a positive weight π_i, defined by

$$\pi_i := \exp \left(\frac{-d(\mathbb{1}, Z_i)^2}{2\nu^2} \right) , \quad (15)$$

where d is the *cosine distance* and $\nu > 0$ is a bandwidth parameter. The intuition behind these weights is that X_i can be far away from ξ if many words are removed (in the most extreme case, $s = d$, all the words from ξ are removed). In that case, z_i has mostly 0 components, and is far away from $\mathbb{1}$ from the point of view of the cosine distance.

Surrogate model. The next step is to train a surrogate model on $Z_1, \ldots, Z_n$, trying to match the responses $Y_i := f(X_i)$. In the default implementation of LIME, this model is linear and is obtained by weighted ridge regression. Formally, LIME outputs

$$\widehat{\beta}_n^\lambda \in \arg\min_{\beta \in \mathbb{R}^{d+1}} \left\{ \sum_{i=1}^{n} \pi_i (y_i - \beta^\top z_i)^2 + \lambda \|\beta\|^2 \right\}, \quad (16)$$

where $\lambda > 0$ is a regularization parameter. We call the components of $\widehat{\beta}_n^\lambda$ the *interpretable coefficients*, the 0th coordinate in our notation is by convention the intercept.

5.2. Limit explanations for our model

Under mild assumptions, Mardaoui & Garreau (Theorem 1, 2021) show that LIME's coefficients converge to *limit coefficients* β^∞. Namely, the number of perturbed samples n is large, the penalization in Eq. (16) is not too strong (which is the case by default), and the bandwidth ν is also large. The expression for the limit coefficient associated to word j,

$$\beta_j^\infty = 3\mathbb{E}[f(X)|j \notin S] - \frac{3}{d} \sum_k \mathbb{E}[f(X)|k \notin S] , \quad (17)$$
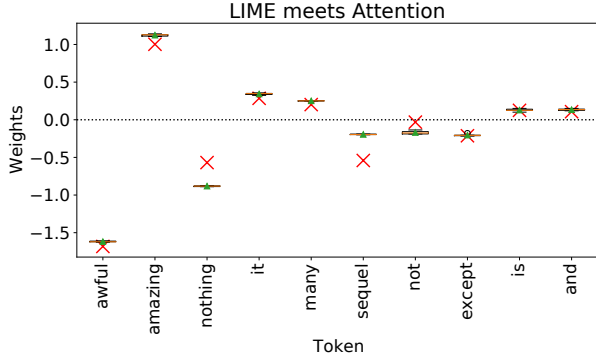
Figure 4. Illustration of the accuracy of Eq. (19). The boxplots show the results from 5 runs of LIME with default parameters, while the red crosses indicate the predictions given by Theorem 5.1. The document ξ contains $T = 99$ tokens and $d = 71$ distinct words and is classified as a negative review. Note that Theorem 5.1 holds true even with $T \neq d$ for reasonable word multiplicities, as discussed in Section 6.

can then be computed (exactly or approximately) as a function of the parameters of the model to gain some precise insights on the behavior of LIME in this situation. This computation is the main result of this section.

Before stating Theorem 5.1, we need some additional notation. Indexing by h will denote the quantity corresponding to the [UNK] token. In particular, $k_{h,t} := W_k h + W_k W_p(t) \in \mathbb{R}^{d_{\text{att}}}$ is the key vector associated to the [UNK] token at position $t \in [T_{\max}]$. For any $t \in [T_{\max}]$, we further define

$$g_{h,t} := \exp\left(q^\top k_{h,t}/\sqrt{d_{\text{att}}}\right), \qquad (18)$$

and

$$\alpha_{h,t} := \frac{g_{h,t}}{\sum_u g_{h,u}}.$$

We note that $\alpha_{h,t}$ can be seen as the attention corresponding to the [UNK] token at position t from the perspective of the query associated to the [CLS] token. Finally, set $v_{h,t} := W_v(h + W_p(t))$. We have:

Theorem 5.1 (LIME meets attention). *Assume that $d = T = T_{\max}^\varepsilon$, with $\varepsilon \in (0, 1)$. Assume further that there exist positive constants $0 < c < C$ such that, as $T \to +\infty$, for all $t \in [T_{\max}]$, $\max(|v_t|, |v_{h,t}|) \leqslant C$, and $c \leqslant \min(g_t, g_{h,t}) \leqslant C$. Then*

$$\beta_j^\infty = \frac{3}{2K} \sum_{i=1}^K \sum_{t=1}^{T_{\max}} W_\ell^{(i)}\left(\alpha_t^{(i)} v_t^{(i)} - \alpha_{h,t}^{(i)} v_{h,t}^{(i)}\right) \mathbb{1}_{\xi_t = j}$$

$$+ \mathcal{O}\left(T_{\max}^{(2-\varepsilon) \vee 3/2}\right). \qquad (19)$$

Theorem 5.1 is proved in Section B. The challenging part of the proof is to derive good approximations for Eq. (17),

since the model we consider, although single-layered, is highly non-linear. Note that Theorem 5.1 relies on the assumption that all tokens are distinct. While we conjecture that this assumption can be relaxed (as discussed in Section 6), it is necessary for a rigorous comparison between attention and LIME explanations. However, our experiments were conducted without assuming all tokens are distinct. We nevertheless observe a very good match between our theoretical approximation and the empirical outputs of LIME with default parameters, which we illustrate in Figure 4 on a particular example, and in a more quantitative way in Appendix B.

We now make a few comments. (i) it is clear from Eq. (19) that LIME explanations are **quite different** from gradient-based explanations (Eq. (14)), with the exception of the leading term (which is proportional to $\alpha_t W_\ell v_t$) which we recognize in both expressions. (ii) one can see that **LIME explanations are approximately an affine transformation of the** $\alpha_t^{(i)}$. (iii) as it is the case for gradient-based explanation, there is a major difference with plain attention-based explanations: the last layer comes into account in the explanation. To put it plainly, let us assume that $K = 1$ and that $W_\ell v_t = 0$, **the influence of α_t disappears whatever its value**. We see this as an advantage for LIME, since this situation corresponds to the model killing the influence of token t in later stages, although a positive attention score is given. (iv) from Eq. (19), we see that **LIME explanations will be near zero** whenever $\alpha_t^{(i)} v_t^{(i)} \approx \alpha_{h,t}^{(i)} v_{h,t}^{(i)}$, a scenario in which the **attention $\times$ value of head i given to token t if comparable to that of the attention given to the [UNK] token**. This makes a lot of sense, while calling for a careful choice of embedding for the replacement token.

6. Limitation

The main differences between the model described in Section 2 and practical architectures are the following: (i) number of layers, (ii) skip connections, (iii) non-linearities. (i): we only consider a single layer, which already brings non-linearity while giving a quite challenging analysis and a very good performance in practice for our task. (ii): we do not consider skip connections since we did not observe an increase in performance while adding them, but our analysis can easily be adapted to this setting since this operation is linear. (iii): in line with typical theoretical works (Gunasekar et al., 2018), we refrain from incorporating additional non-linearities; specifically, we do not introduce a ReLU activation or any of its variants.

The primary limitation of the present analysis lies in its focus on a single-layer model. This facilitates a deep theoretical exploration, yet its applicability to more intricate architectures may not be direct. The extension of this analysis to a multi-layer architecture introduces additional theoretical

challenges. In the context of multi-layer transformers, approximation errors have the potential to accumulate and intensify as they traverse through the network. Moreover, the assumptions applicable to a single layer may not necessarily hold true for deeper networks, especially those incorporating non-linearities such as ReLU activations. However, even for simple architectures, many questions remain unresolved, and the interpretability of these models has not been thoroughly studied in a formal manner.

Several theoretical studies on transformers share these limitations. For example, Jelassi et al. (2022) elucidates how Vision Transformers discern spatial patterns within a single-layer, single-head architecture. Similarly, Tarzanagh et al. (2023) establishes the correspondence between the optimization geometry of self-attention and an SVM problem. Von Oswald et al. (2023) suggests that training Transformers with a specific objective can induce a form of meta-learning, exemplified on a linear single-layer model. Additionally, Edelman et al. (2022) investigate transformers in time series forecasting, showcasing their superiority over traditional methods by accurately capturing temporal dependencies. In a bid to enhance transformer efficiency and scalability, Fu et al. (2023) introduce sparse attention mechanisms and efficient training strategies by formalizing single-layer transformers. Furthermore, Makkuva et al. (2024) tackle transformer interpretability, developing tools to visualize and comprehend attention patterns within the models, aiming to bridge the gap between high performance and the imperative for transparency in critical applications. In Appendix E, we report experiments on a multi-layer transformer.

We conclude this section by discussing limitations of Theorem 5.1. First, as in previous work, the approximation is only true for large document and window size. This comes without surprise, but we note that the results are experimentally satisfying for documents which are a few dozen tokens long. Second, we assume in Theorem 5.1 that all tokens are distinct. This assumption, though a simplification, allows for a rigorous formalization of LIME's behavior using its default parameters as defined in the official implementation. Experimentally, Eq. (19) holds for documents containing repeated words. We validated Theorem 5.1 disregarding this assumption, by computing the norm-2 error between the official LIME weights and our approximation (as illustrated in Figure 4): the average error over the full test set (described in Appendix F) is 0.808, with a standard deviation of 0.219. From a theoretical aspect, we conjecture that this assumption can be relaxed if the maximal multiplicity of tokens is small relative to T. If many tokens are identical, this is no longer true: consider, for instance, the extreme case of two groups of identical tokens. We delve into this topic in more detail in Appendix B.1.

7. Conclusion and future work

In this paper, we offered a theoretical analysis on how post-hoc explanations relates to a single-layer multi-head attention-based network. Our work contributes to the ongoing debate in this area by providing exact and approximate expressions for post-hoc explanations on such model. Through these expressions, we were able to highlight the fundamental differences between attention-based, gradient-based, and perturbation-based explanations. This deeper understanding not only enriches the ongoing discourse surrounding interpretability but also offers valuable insights for practitioners and researchers navigating the complexities of transformers' interpretation.

It is crucial to acknowledge that the quest for perfect explanations remains elusive; no single method has emerged as entirely satisfactory. However, it is clear that current models employ attention scores in a non-intuitive manner to arrive at the final prediction. In particular, these scores go through a series of further transformations, which is ignored when looking solely at attention scores. These scores also always provide a positive explanation, in contrast to (most) perturbation-based and gradient-based approaches. For these reasons, we believe that they can extract more valuable insights than a mere examination of attention weights. This finding aligns with the assertions made by Bastings & Filippova (2020).

As future work, we plan to broaden the scope of our analysis by extending our investigations to diverse range of post-hoc interpretability methods, including Anchors, thus understanding model explanations across different methodologies. We also would like to obtain similar statements (connecting explanations to the parameters of the model) for more complicated architectures, including skip connections, additional non-linearities, and multi-layer models, enabling us to discern the relationship between model parameters and different explanations. Addionally, there is some interplay between the sampling mechanism of perturbation-based methods (often replacing at the word level) and the tokenizer used by the model (tokens are often subwords) which we would like to understand better. Lastly, we emphasize that our focus in this paper has been on text classification. This choice allows us to capitalize on well-established, and broadly studied post-hoc explainers and conduct a thorough theoretical analysis based on this specific domain. However, we intend to expand the scope of applications for our analysis. Specifically, we remark that our study focused on token-level explanations. Moving forward, we intend to extend our findings beyond text models to encompass other domains, such as computer vision.

Acknowledgements

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Arras, L., Osman, A., Müller, K.-R., and Samek, W. Evaluating recurrent neural network explanations. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 113–126, 2019.

Atanasova, P., Simonsen, J. G., Lioma, C., and Augenstein, I. A diagnostic study of explainability techniques for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 3256–3274, 2020.

Bahdanau, D., Cho, K. H., and Bengio, Y. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations (ICLR)*, 2015.

Bastings, J. and Filippova, K. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pp. 149–155, 2020.

Bibal, A., Cardon, R., Alfter, D., Wilkens, R., Wang, X., François, T., and Watrin, P. Is attention explanation? an introduction to the debate. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3889–3900, 2022.

Blei, D. M., Ng, A. Y., and Jordan, M. I. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan): 993–1022, 2003.

Bodria, F., Giannotti, F., Guidotti, R., Naretto, F., Pedreschi, D., and Rinzivillo, S. Benchmarking and survey of explanation methods for black box models. *Data Mining and Knowledge Discovery*, pp. 1–60, 2023.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.

Brunner, G., Liu, Y., Pascual, D., Richter, O., Ciaramita, M., and Wattenhofer, R. On identifiability in transformers. In *8th International Conference on Learning Representations (ICLR)*, 2020.

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.

Clark, K., Khandelwal, U., Levy, O., and Manning, C. D. What does bert look at? an analysis of bert's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 276–286, 2019.

Cui, H., Behrens, F., Krzakala, F., and Zdeborová, L. A phase transition between positional and semantic learning in a solvable model of dot-product attention. *arXiv preprint arXiv:2402.03902*, 2024.

Denil, M., Demiraj, A., and De Freitas, N. Extraction of salient sentences from labelled documents. *arXiv preprint arXiv:1412.6815*, 2014.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

Edelman, B. L., Goel, S., Kakade, S., and Zhang, C. Inductive biases and variable creation in self-attention mechanisms. In *International Conference on Machine Learning*, pp. 5793–5831. PMLR, 2022.

Ethayarajh, K. and Jurafsky, D. Attention flows are Shapley value explanations. *arXiv preprint arXiv:2105.14652*, 2021.

Fu, H., Guo, T., Bai, Y., and Mei, S. What can a Single Attention Layer Learn? A Study Through the Random Features Lens. In *Advances in Neural Information Processing Systems*, volume 36, pp. 11912–11951, 2023.

Galassi, A., Lippi, M., and Torroni, P. Attention in natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 32(10):4291–4308, 2020.

Gunasekar, S., Lee, J. D., Soudry, D., and Srebro, N. Implicit bias of gradient descent on linear convolutional networks. *Advances in neural information processing systems*, 31, 2018.

Hase, P., Xie, H., and Bansal, M. The out-of-distribution problem in explainability and search methods for feature importance explanations. *Advances in Neural Information Processing Systems*, 34:3650–3666, 2021.

Huang, Z., Xu, W., and Yu, K. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

Jain, S. and Wallace, B. C. Attention is not explanation. In *Proceedings of NAACL-HLT*, pp. 3543–3556, 2019.

Jelassi, S., Sander, M., and Li, Y. Vision transformers provably learn spatial structure. *Advances in Neural Information Processing Systems*, 35:37822–37836, 2022.

Kobayashi, G., Kuribayashi, T., Yokoi, S., and Inui, K. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7057–7075, 2020.

Li, J., Chen, X., Hovy, E., and Jurafsky, D. Visualizing and understanding neural models in nlp. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 681–691, 2016.

Li, Y., Li, Y., and Risteski, A. How do transformers learn topic structure: Towards a mechanistic understanding. In *International Conference on Machine Learning*, pp. 19689–19729. PMLR, 2023.

Linardatos, P., Papastefanopoulos, V., and Kotsiantis, S. Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy*, 23(1):18, 2021.

Lopardo, G., Precioso, F., and Garreau, D. A Sea of Words: An In-Depth Analysis of Anchors for Text Data. In *International Conference on Artificial Intelligence and Statistics*, 2023a.

Lopardo, G., Precioso, F., and Garreau, D. Faithful and Robust Local Interpretability for Textual Predictions. *arXiv preprint arXiv:2311.01605*, 2023b.

Lundberg, S. M. and Lee, S.-I. A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*, 30:4765–4774, 2017.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150. Association for Computational Linguistics, 2011.

Makkuva, A. V., Bondaschi, M., Girish, A., Nagle, A., Jaggi, M., Kim, H., and Gastpar, M. Attention with Markov: A framework for principled analysis of transformers via markov chains. *arXiv preprint arXiv:2402.04161*, 2024.

Mardaoui, D. and Garreau, D. An analysis of LIME for text data. In *International Conference on Artificial Intelligence and Statistics*, pp. 3493–3501. PMLR, 2021.

Marques-Silva, J. and Ignatiev, A. Delivering trustworthy ai through formal xai. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 12342–12350, 2022.

Mylonas, N., Mollas, I., and Tsoumakas, G. An attention matrix for every decision: Faithfulness-based arbitration among multiple attention-based interpretations of transformers in text classification. *Data Mining and Knowledge Discovery*, pp. 1–26, 2023.

Neely, M., Schouten, S. F., Bleeker, M. J. R., and Lucic, A. Order in the court: Explainable AI methods prone to disagreement. *arXiv preprint arXiv:2105.03287*, 2021.

Neely, M., Schouten, S. F., Bleeker, M., and Lucic, A. A Song of (Dis)agreement: Evaluating the Evaluation of Explainable Artificial Intelligence in Natural Language Processing. In *HHAI2022: Augmenting Human Intellect*, pp. 60–78. IOS Press, 2022.

Phuong, M. and Hutter, M. Formal algorithms for transformers. *arXiv preprint arXiv:2207.09238*, 2022.

Poerner, N., Schütze, H., and Roth, B. Evaluating neural network explanation methods using hybrid documents and morphosyntactic agreement. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 340–350, 2018.

Ribeiro, M. T., Singh, S., and Guestrin, C. "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*, pp. 1135–1144, 2016.

Ribeiro, M. T., Singh, S., and Guestrin, C. Anchors: High-precision model-agnostic explanations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Sak, H., Senior, A., and Beaufays, F. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *arXiv preprint arXiv:1402.1128*, 2014.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Schützenberger, M. P. On context-free languages and pushdown automata. *Information and control*, 6(3):246–264, 1963.

Serrano, S. and Smith, N. A. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2931–2951, 2019.

Shapley, L. S. A value for n-person games. *Contributions to the Theory of Games, number 28 in Annals of Mathematics Studies, pages 307–317*, II, 1953.

Slack, D., Hilgard, S., Jia, E., Singh, S., and Lakkaraju, H. Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 180–186, 2020.

Sun, K. and Marasović, A. Effective attention sheds light on interpretability. In *Findings of the Association for Computational Linguistics (ACL-IJCNLP)*, pp. 4126–4135, 2021.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328, 2017.

Tarzanagh, D. A., Li, Y., Thrampoulidis, C., and Oymak, S. Transformers as support vector machines. In *NeurIPS 2023 Workshop on Mathematics of Modern Machine Learning*, 2023.

Thorne, J., Vlachos, A., Christodoulopoulos, C., and Mittal, A. Generating token-level explanations for natural language inference. *arXiv preprint arXiv:1904.10717*, 2019.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

Vig, J. and Belinkov, Y. Analyzing the structure of attention in a transformer language model. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 63–76, 2019.

Von Oswald, J., Niklasson, E., Randazzo, E., Sacramento, J., Mordvintsev, A., Zhmoginov, A., and Vladymyrov, M. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pp. 35151–35174. PMLR, 2023.

Weiss, G., Goldberg, Y., and Yahav, E. Thinking like transformers. In *International Conference on Machine Learning*, pp. 11080–11090. PMLR, 2021.

Wen, K., Li, Y., Liu, B., and Risteski, A. Transformers are uninterpretable with myopic methods: a case study with bounded dyck grammars. *Advances in Neural Information Processing Systems*, 36, 2024.

Wiegreffe, S. and Pinter, Y. Attention is not not Explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 11–20, 2019.

Appendix for the paper
Attention Meets Post-hoc Interpretability: A Mathematical Perspective

Organization of the Appendix. We start by providing proofs for the theoretical results presented in the paper. We prove Theorems 4.1 and 5.1 in Sections A and B, respectively. Sections C and D collect additional technical details crucial for the proofs. Finally, in Section F, we detail the model employed for our experiments. For further information, the training and experimental code are available at `https://github.com/gianluigilopardo/attention_meets_xai`.

A. Proof of Theorem 4.1

In this section, we show how to compute the gradient of f with respect to the embedding e_t, $t \in [T]$. By linearity, we can focus on one head f_i, $i \in [K]$, and thus we momentarily drop the i superscripts. Let us start by computing the gradients of the key, query, and value vectors k_t, q_t, and v_t (Eqs. (5), (6), and (7)). For any $t \in [T]$, one has

$$\nabla_{e_t} k_t = \nabla_{e_t} (W_k e_t) = W_k (\nabla_{e_t} e_t) = W_k^\top \in \mathbb{R}^{d_e \times d_{att}}, \tag{20}$$

$$\nabla_{e_t} q_t = \nabla_{e_t} (W_q e_t) = W_q (\nabla_{e_t} e_t) = W_q^\top \in \mathbb{R}^{d_e \times d_{att}}, \tag{21}$$

and

$$\nabla_{e_t} v_t = \nabla_{e_t} (W_v e_t) = W_v^\top \in \mathbb{R}^{d_e \times d_{out}}. \tag{22}$$

Therefore, the gradient of the attention α_t as defined in Eq. (8), for any $t \in [T]$,

$$\nabla_{e_t} \alpha_t = \nabla_{e_t} \left(\frac{\exp\left(q^\top k_t / \sqrt{d_{att}}\right)}{\sum_{u=1}^{T_{max}} \exp\left(q^\top k_u / \sqrt{d_{att}}\right)} \right)$$

$$= \frac{\alpha_t}{\sqrt{d_{att}}} \left((W_k^\top q) - \sum_{s=1}^{T_{max}} \alpha_s (W_k^\top q) \right) \quad \in \mathbb{R}^{d_e}.$$

The situation is similar if we look at another attention coefficient: let $s \neq t$, then

$$\nabla_{e_t} \alpha_s = \nabla_{e_t} \left(\frac{\exp\left(q^\top k_s / \sqrt{d_{att}}\right)}{\sum_{u=1}^{T_{max}} \exp\left(q^\top k_u / \sqrt{d_{att}}\right)} \right)$$

$$= \frac{-\alpha_u \alpha_t}{\sqrt{d_{att}}} (W_k^\top q).$$

Finally, we can compute the gradient of $\tilde{v}$ as

$$\nabla_{e_t} \tilde{v} = \nabla_{e_t} \left(\sum_{u=1}^{T_{max}} \alpha_u v_u \right)$$

$$= \nabla_{e_t}(\alpha_t) v_t + \alpha_t (\nabla_{e_t} v_t) + \sum_{s \neq t} (\nabla_{e_t} \alpha_s) v_s$$

$$= \frac{1}{\sqrt{d_{att}}} (W_k^\top q)(\alpha_t - \alpha_t^2) v_t + \alpha_t W_v^\top + \sum_{s \neq t} \frac{-\alpha_u \alpha_t}{\sqrt{d_{att}}} (W_k^\top q)$$

$$\nabla_{e_t} = \frac{\alpha_t}{\sqrt{d_{att}}} \left(v_t - \sum_{s=1}^{T_{max}} \alpha_s v_s \right) (W_k^\top q) + \alpha_t W_v^\top.$$
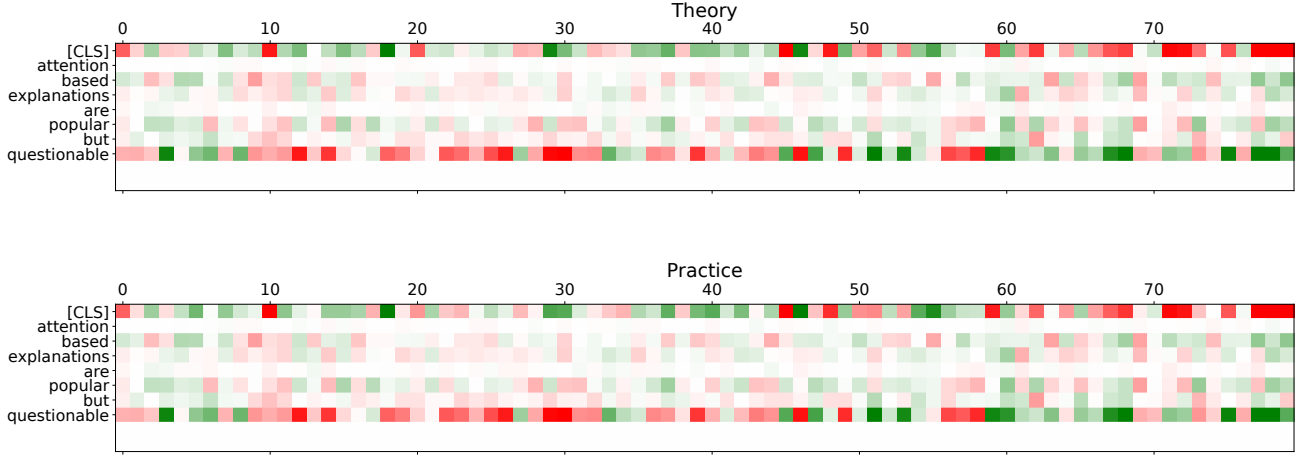
Figure 5. Illustration of the accuracy of Theorem 4.1. Here, for illustrative purpose, $d_e = 80$.

Finally, since $f(x) = W_\ell \tilde{v}$, we deduce Eq. (14) from the last display, multiplying by $W_\ell^\top$ and averaging. $\qquad\square$

Theorem 4.1 is also true in practice, as illustrated in Figure 5.

B. Proof of Theorem 5.1

Preliminaries. The key idea of this proof is to leverage Eq. (17) and find a good approximation for the conditional expectations involved. Looking closer at Eq. (17), we first notice that, by linearity, we can focus on the limit coefficients associated to a single head. Thus we drop the i indexation in this proof.

Now let us recall that S is the random subset of words from the dictionary being removed when generating X. Our first key observation is that X **has random token embeddings** E_t. More precisely, Eq. (2) becomes

$$\forall t \in [T_{\max}], \quad E_t := e_t \mathbb{1}_{\xi_t \notin S} + (h + W_p(t)) \mathbb{1}_{\xi_t \in S} . \tag{23}$$

We note that E_t for $t > T$ is actually not random (LIME does not perturb outside of ξ), but this will be of no consequence. In turn, keys and queries are modified, that is, Eqs. (5) and (6) become, respectively,

$$\forall t \in [T_{\max}], \quad K_t := k_t \mathbb{1}_{\xi_t \notin S} + W_k(h + W_p(t)) \mathbb{1}_{\xi_t \in S} , \tag{24}$$

and

$$\forall t \in [T_{\max}], \quad Q_t := q_t \mathbb{1}_{\xi_t \notin S} + W_q(h + W_p(t)) \mathbb{1}_{\xi_t \in S} . \tag{25}$$

The attention coefficients associated to the [CLS] token also become random. In analogous fashion to Eq. (18), let us define

$$\forall u \in [T_{\max}], \quad G_u := \exp\left(q^\top K_u / \sqrt{d_{\mathrm{att}}}\right) ,$$

where we recall that $q \in \mathbb{R}^{d_{\mathrm{att}}}$ is the query vector associated to the [CLS] token. Taking dot product and exponential, and noting that the indicator functions concern disjoint events, we see that

$$\forall u \in [T_{\max}], \quad G_u = g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S} , \tag{26}$$

where we let $g_u := \exp\left(q^\top k_u / \sqrt{d_{\mathrm{att}}}\right)$ and $g_{h,t} = \exp\left(q^\top k_{h,t} / \sqrt{d_{\mathrm{att}}}\right)$ as in Eq. (18). Then, with this notation in hand, we define the random attention coefficient associated to token t by

$$\forall t \in [T_{\max}], \quad A_t := \frac{G_t}{\sum_{u=1}^{T_{\max}} G_u} . \tag{27}$$

Finally, value vectors are also random in this setting. Namely,

$$\forall t \in [T_{\max}], \quad V_t := v_t \mathbb{1}_{\xi_t \notin S} + v_{h,t} \mathbb{1}_{\xi_t \in S} , \tag{28}$$

where we recall that $v_{h,t} = W_v(h + W_p(t))$.

14

Reduction to key computation. Looking at Eq. (17), and now with appropriate notation, we need to compute

$$\mathbb{E}\left[f(X)|\ell \notin S\right] = \mathbb{E}\left[\sum_{t=1}^{T_{\max}} A_t V_t \middle| \ell \notin S\right]$$

for all $\ell \in [d]$. Again by linearity, one can focus on the computation of $\mathbb{E}\left[A_t V_t|\ell \notin S\right]$. The following result gives an approximation of this quantity when both $T_{\max}$ and d are large:

Proposition B.1 (Approximated conditional expectation). *Assume that $d = T$. Assume further that there exist positive constants $0 < c < C$ such that, as $T \to +\infty$, for all $t \in [T_{\max}]$, $\max(|v_t|, |v_{h,t}|) \leqslant C$, and $c \leqslant \min(g_t, g_{h,t}) \leqslant C$. Then, for any $t \in [T_{\max}]$, if $\xi_t = \ell$,*

$$\mathbb{E}\left[A_t V_t | \ell \notin S\right] = \frac{1}{d}\sum_{s=1}^{d-1} \frac{g_t v_t}{\left(1 - \frac{s}{d-1}\right)\sum_u g_u + \frac{s}{d-1}\sum_u g_{h,u}} + \mathcal{O}\left(T_{\max}^{-3/2}\right), \tag{29}$$

and otherwise

$$\mathbb{E}\left[A_t V_t | \ell \notin S\right] = \frac{1}{d}\sum_{s=1}^{d-1} \frac{\left(1 - \frac{s}{d-1}\right)g_t v_t + \frac{s}{d-1}g_{h,t}v_{h,t}}{\left(1 - \frac{s}{d-1}\right)\sum_u g_u + \frac{s}{d-1}\sum_u g_{h,u}} + \mathcal{O}\left(T_{\max}^{-3/2}\right). \tag{30}$$

The proof of Proposition B.1 is deferred to Section C. From Eqs. (29) and (30), coming back to Eq. (17), we deduce that the jth limit coefficient associated to $A_t V_t$ is approximately equal to

$$\frac{3}{d}\sum_{s=1}^{d-1} \frac{\frac{s}{d-1}(g_t v_t - g_{h,t}v_{h,t})}{\left(1 - \frac{s}{d-1}\right)\sum_u g_u + \frac{s}{d-1}\sum_u g_{h,u}} + \mathcal{O}\left(T_{\max}^{-3/2}\right). \tag{31}$$

The derivative of the mapping

$$x \mapsto \frac{x}{(1-x)\sum_u g_u + x\sum_u g_{h,u}}$$

is given by

$$x \mapsto \frac{\sum_u g_u}{(x(\sum_u g_{h,u} - \sum_u g_u) + \sum_u g_u)^2}.$$

On $[0, 1]$, under our assumptions, the last display is uniformly bounded by $\mathcal{O}\left(T_{\max}^{-1}\right)$ in absolute value. Thus, by standard Riemann sum approximation, the last display is

$$3(g_t v_t - g_{h,t}v_{h,t})\int_0^1 \frac{x\mathrm{d}x}{(1-x)\sum_u g_u + x\sum_u g_{h,u}} + \mathcal{O}\left(T_{\max}^{-3/2}\right).$$

Let us recall that, if $T < u \leqslant T_{\max}$, $g_u = g_{h,u}$. Therefore, $\sum_u g_{h,u} - \sum_u g_u = \mathcal{O}(T) = \mathcal{O}(T_{\max}^\varepsilon)$, and $(\sum_u g_{h,u} - \sum_u g_u)/\sum_u g_u = \mathcal{O}\left(T_{\max}^{\varepsilon-1}\right)$. Therefore, according to Lemma D.2, the integral in the last display can be well approximated by

$$\frac{1}{\sum_u g_u}\cdot\left(\frac{1}{2} + \mathcal{O}\left(\frac{\sum_u g_{h,u} - \sum g_u}{\sum g_u}\right)\right) = \frac{1}{2\sum g_u} + \mathcal{O}\left(T_{\max}^{\varepsilon-2}\right).$$

The same reasoning shows that, whenever $\xi_t \neq j$, the approximation is zero (with the same precision in the error). Thus, by linearity (over the tokens and the model), we obtain the statement of Theorem 5.1. $\square$

B.1. Discussion on Theorem 5.1

A theoretical limitation of Theorem 5.1 it the assumption of distinct tokens. This assumption, while technically a simplification, enables a rigorous formalization of LIME's behavior using its default parameters as defined in the official implementation. We conducted quantitative experiments that disregarded this assumption, and the results still hold. We empirically validate the accuracy of Theorem 5.1 by computing the norm-2 error between the LIME weights from the official implementation (available on Github at https://github.com/marcotcr/lime) and our approximation. The average norm-2 error, computed over the full test set (see Section F), is 0.808, with a standard deviation of 0.219.

The primary challenge in proving a formal result that allows for repetitions lies in the use of Lemma D.1. The key intuition in the current proof mechanism is that if only one element of the denominator of A_t varies randomly, this has a minimal overall effect on the entire denominator, given that it has $T \gg 1$ terms. However, if many tokens are identical, this is no longer true, and it can result in high variance (consider, for instance, the extreme case of two groups of identical tokens), which prevents us from using Lemma D.1. Nevertheless, we conjecture that if the tokens are not distinct, but the maximal multiplicity of tokens is small relative to T, our findings hold true (and this is empirically true).

On a more practical note, we highlight that by default, LIME-text perturbs input data by removing all occurrences of individual words or characters (bow=True, *i.e.*, bag-of-words), and this is the subject of our study. However, if the underlying model uses word location (as in our classifier), a possibility is to set bow=False (as recommended in their notebook), so that any occurrence of the same word is considered a distinct token. By using this option, the same applies in Theorem 5.1: the same words in different parts of the text are considered different tokens.

C. Proof of Proposition B.1

Sketch of the proof. Essentially, assuming for a second that V_t is constant, the crux of the result is to compute the (approximate) expectation of A_t, which is defined as the ratio of positive quantities (Eq. (27)). Since the denominator is quite large, one can use Lemma D.1 and approximate the expected ratio by the ratio of expectation. This works only if, concurrently, the variance is not too high, which is guaranteed by Lemma D.3. Lemmas D.1 and D.3 are stated and proved in Section D.

Proof of Proposition B.1. We first write

$$\mathbb{E}\left[A_t V_t | \ell \notin S\right] = \mathbb{E}\left[\frac{G_t V_t}{\sum_{u=1}^{T_{\max}} G_u}\bigg| \ell \notin S\right] \qquad \text{(Eqs. (27) and (28))}$$

$$= \mathbb{E}\left[\frac{g_t v_t \mathbb{1}_{\xi_t \notin S} + g_{h,t} v_{h,t} \mathbb{1}_{\xi_t \in S}}{\sum_{u=1}^{T_{\max}}\{g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S}\}}\bigg| \ell \notin S\right] \qquad \text{(Eq. (26))}$$

$$= \frac{1}{d}\sum_{s=0}^{d-1} \mathbb{E}_s\left[\frac{g_t v_t \mathbb{1}_{\xi_t \notin S} + g_{h,t} v_{h,t} \mathbb{1}_{\xi_t \in S}}{\sum_{u=1}^{T_{\max}}\{g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S}\}}\bigg| \ell \notin S\right]. \qquad \text{(law of total expectation)}$$

Note that there is no $s = d$ term in the last display, since d removals is incompatible with $\ell \notin S$. Let us set $s \in [d-1]$. Define $X := g_t v_t \mathbb{1}_{\xi_t \notin S} + g_{h,t} v_{h,t} \mathbb{1}_{\xi_t \in S}$ and $Y := \sum_{u=1}^{T_{\max}}\{g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S}\}$. Under our assumptions, X is clearly bounded while Y has order $T_{\max}$. Thus the hypotheses of Lemma D.1 are satisfied with $n = T_{\max}$. Moreover, Lemma D.3 guarantees that $\text{Var}_s(Y \mid \ell \notin S) = \mathcal{O}(T_{\max})$. From Lemma D.1, we deduce that

$$\mathbb{E}_s\left[\frac{g_t v_t \mathbb{1}_{\xi_t \notin S} + g_{h,t} v_{h,t} \mathbb{1}_{\xi_t \in S}}{\sum_{u=1}^{T_{\max}}\{g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S}\}}\bigg| \ell \notin S\right] = \frac{\mathbb{E}_s\left[g_t v_t \mathbb{1}_{\xi_t \notin S} + g_{h,t} v_{h,t} \mathbb{1}_{\xi_t \in S} | \ell \notin S\right]}{\mathbb{E}_s\left[\sum_{u=1}^{T_{\max}}\{g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S}\}\big| \ell \notin S\right]} + \mathcal{O}\left(T_{\max}^{-3/2}\right). \qquad (32)$$

Let us assume from now on that $\xi_t = \ell$ (the case $\xi_t \neq \ell$ is similar). Then

$$\mathbb{E}_s\left[g_t v_t \mathbb{1}_{\xi_t \notin S} + g_{h,t} v_{h,t} \mathbb{1}_{\xi_t \in S} | \ell \notin S\right] = g_t v_t, \qquad (33)$$

and for all $u \neq t$,

$$\mathbb{E}_s\left[g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S} | \ell \notin S\right] = g_u \mathbb{P}_s\left(\xi_u \notin S \mid \ell \notin S\right) + g_{h,u} \mathbb{P}_s\left(\xi_u \in S \mid \ell \notin S\right).$$

Since we assumed distinct tokens ($d = T$), Lemma D.5 yields

$$\mathbb{E}_s\left[g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S} | \ell \notin S\right] = \left(1 - \frac{s}{d-1}\right) g_u + \frac{s}{d-1} g_{h,u}. \qquad (34)$$

Injecting Eqs. (33) and (34) into Eq. (32), we obtain

$$\mathbb{E}_s\left[\frac{g_t v_t \mathbb{1}_{\xi_t \notin S} + g_{h,t} v_{h,t} \mathbb{1}_{\xi_t \in S}}{\sum_{u=1}^{T_{\max}}\{g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S}\}}\bigg| \ell \notin S\right] = \frac{g_t v_t}{\left(1 - \frac{s}{d-1}\right)\sum_u g_u + \frac{s}{d-1}\sum_u g_{h,u} + \frac{s}{d-1}(g_t - g_{h,t})} + \mathcal{O}\left(T_{\max}^{-3/2}\right).$$

16

Under our assumptions, $\frac{s}{d-1}(g_t - g_{h,t})$ is $\mathcal{O}(1)$, whereas the remainder of the denominator is of order at least $T_{\max}$. We deduce that

$$\mathbb{E}_s \left[\left. \frac{g_t v_t \mathbb{1}_{\xi_t \notin S} + g_{h,t} v_{h,t} \mathbb{1}_{\xi_t \in S}}{\sum_{u=1}^{T_{\max}} \{g_u \mathbb{1}_{\xi_u \notin S} + g_{h,u} \mathbb{1}_{\xi_u \in S}\}} \right| \ell \notin S \right] = \frac{g_t v_t}{\left(1 - \frac{s}{d-1}\right) \sum_u g_u + \frac{s}{d-1} \sum_u g_{h,u}} + \mathcal{O}\left(T_{\max}^{-3/2}\right). \tag{35}$$

We deduce the result coming back to the initial decomposition. $\qquad\square$

D. Technical results

Lemma D.1 (Expected ratio). *Let X and Y be two random variables with finite variance. Assume that there exist two positive constants c and C such that $|X| \leqslant C$ and $cn \leqslant Y \leqslant Cn$ a.s. Then*

$$\left| \mathbb{E}\left[\frac{X}{Y}\right] - \frac{\mathbb{E}[X]}{\mathbb{E}[Y]} \right| \leqslant \frac{C \operatorname{Var}(Y)}{c^3 n^3} + \frac{C^2 \sqrt{\operatorname{Var}(Y)}}{c^2 n^2}.$$

Proof. Set $\psi : \mathbb{R}_+^2 \to \mathbb{R}$ defined as $\psi(x,y) := x/y$. Multivariate Taylor expansion at order 1 with integral remainder for an arbitrary $(x_0, y_0) \in \mathbb{R}_+^2$ yields

$$\psi(x,y) = \psi(x_0, y_0) + (x - x_0)\partial_x \psi(x_0, y_0) + (y - y_0)\partial_y \psi(x_0, y_0)$$
$$+ \frac{2}{2!}(x - x_0)^2 \int_0^1 (1-t)\partial_{xx}\psi((x_0, y_0) + t(x - x_0, y - y_0))\mathrm{d}t$$
$$+ \frac{2}{1!1!}(x - x_0)(y - y_0) \int_0^1 (1-t)\partial_{xy}\psi((x_0, y_0) + t(x - x_0, y - y_0))\mathrm{d}t$$
$$+ \frac{2}{2!}(y - y_0)^2 \int_0^1 (1-t)\partial_{yy}\psi((x_0, y_0) + t(x - x_0, y - y_0))\mathrm{d}t.$$

Let us focus on the remainder (the last three lines of the previous display). Since $\partial_{xx}\psi = 0$, $\partial_{xy}\psi = -1/y^2$, and $\partial_{yy}\psi = 2x/y^3$, we are left with

$$-2(x - x_0)(y - y_0) \int_0^1 \frac{(1-t)\mathrm{d}t}{(y_0 + t(y - y_0))^2} + 2(y - y_0)^2 \int_0^1 \frac{(1-t)(x_0 + t(x - x_0))\mathrm{d}t}{(y_0 + t(y - y_0))^3},$$

that is,

$$(y - y_0) \cdot \int_0^1 (1-t) \frac{-2(x - x_0)(y_0 + t(y - y_0)) + 2(y - y_0)(x_0 + t(x - x_0))}{(y_0 + t(y - y_0))^3} \mathrm{d}t.$$

One can actually compute this integral, which is

$$(y - y_0) \cdot \frac{x_0 y - x y_0}{y y_0^2} = (y - y_0) \cdot \frac{x_0(y - y_0) - (x - x_0)y_0}{y y_0^2}.$$

Going back to the original expansion, we have proved that

$$\psi(x,y) = \psi(x_0, y_0) + (x - x_0)\partial_x \psi(x_0, y_0) + (y - y_0)\partial_y \psi(x_0, y_0) + (y - y_0) \cdot \frac{x_0(y - y_0) - (x - x_0)y_0}{y y_0^2}. \tag{36}$$

Let us now use Eq. (36) with $x = X$, $y = Y$, $x_0 = \mathbb{E}[X]$, and $y_0 = \mathbb{E}[Y]$, and then take expectation on both sides. We see that the linear term vanishes, and we are left

$$\mathbb{E}\left[\frac{X}{Y}\right] - \frac{\mathbb{E}[X]}{\mathbb{E}[Y]} = \mathbb{E}\left[(Y - \mathbb{E}[Y]) \cdot \frac{\mathbb{E}[X](Y - \mathbb{E}[Y]) - (X - \mathbb{E}[X])\mathbb{E}[Y]}{Y \mathbb{E}[Y]^2}\right].$$

In absolute value, the last display is smaller than

$$\frac{C \operatorname{Var}(Y)}{c^3 n^3} + \frac{C\sqrt{\operatorname{Var}(X)\operatorname{Var}(Y)}}{c^2 n^2},$$

and we deduce the result using Popoviciu's inequality to bound the variance of X. $\qquad\square$

We conclude with a technical result used in approximating an integral appearing in the proof of Theorem 5.1.

Lemma D.2 (Integral approximation). *Let $a > 0$. Then*

$$\int_0^1 \frac{x\,\mathrm{d}x}{1 + ax} = \frac{a - \log(a + 1)}{a^2} = \frac{1}{2} - \frac{a}{3} + \mathcal{O}\left(a^2\right).$$

Proof. Taylor expansion. □

D.1. Conditional variance computations

To use Lemma D.1 in a meaningful way, the variance of the denominator needs to be controlled. We show that this is the case with this next result.

Lemma D.3 (Conditional variance computation). *Let a_i and b_i be two sequences of positive numbers for $i \in [n]$. We set H_S as before. Let $\ell \in [n]$. Then, for all $s \in [n-1]$,*

$$\mathbb{E}_s\left[H_S | \ell \notin S\right] = \frac{n-1-s}{n-1} \sum_i a_i + \frac{s}{n-1} \sum_i b_i + \frac{s}{n-1}(a_\ell - b_\ell),$$

and

$$\mathrm{Var}_s(H_S \mid \ell \notin S) = \frac{ns(n-s-1)}{(n-1)(n-2)} \left[\widehat{\mathrm{Var}}\,(a - b) - \frac{1}{n-1}\left(a_\ell - b_\ell - (\bar{a} - \bar{b})\right)^2 \right],$$

where $\bar{a}$ (resp. $\bar{b}$) denote the empirical mean of a (resp. b).

Lemma D.3 is somewhat remarkable, connecting the variance of the random sum underpinning our problems to the empirical variance of the coefficients. In particular, if we assume that the variance of the summands is $\mathcal{O}(1)$, then $\mathrm{Var}_s(H_S \mid \ell \notin S) = \mathcal{O}(n)$.

Proof. We first write

$$\mathbb{E}_s\left[H_S | \ell \notin S\right] = \mathbb{E}_s\left[\sum_i \{a_i \mathbb{1}_{i \notin S} + b_i \mathbb{1}_{i \in S}\} \middle| \ell \notin S \right]$$
$$= \sum_i a_i\, \mathbb{P}_s\left(i \notin S \mid \ell \notin S\right) + \sum_i b_i\, \mathbb{P}_s\left(i \in S \mid \ell \notin S\right).$$

Taking special care of the case $i = \ell$ in the previous display and using Lemma D.5, we obtain

$$\mathbb{E}_s\left[H_S | \ell \notin S\right] = \frac{n-1-s}{n-1} \sum_{i \neq \ell} a_i + \frac{s}{n-1} \sum_{i \neq \ell} b_i + a_\ell.$$

Rearranging this expression yields the fist statement of the lemma. We now turn to the variance computation. Without loss of generality, we can assume that $b = 0$ since

$$H_S = \sum_i \{a_i \mathbb{1}_{i \notin S} + b_i \mathbb{1}_{i \in S}\} = \sum_i (a_i - b_i) \mathbb{1}_{i \notin S} + \sum_i b_i.$$

Moreover, we notice that

$$\sum_i (a_i + \lambda) \mathbb{1}_{i \notin S} = \sum_i a_i \mathbb{1}_{i \notin S} + \lambda \sum_i \mathbb{1}_{i \notin S} = \sum_i a_i \mathbb{1}_{i \notin S} + (n - s)\lambda.$$

Thus, without loss of generality, we can assume that $\sum_i a_i = 0$. Under this assumption, the expectation is simply

$$\mathbb{E}_s\left[H_S | \ell \notin S\right] = \frac{s}{n-1} a_\ell.$$

We then compute the second raw moment:

$$\mathbb{E}_s\left[H_S^2 \middle| \ell \notin S\right] = \mathbb{E}_s\left[\left(\sum_i a_i \mathbb{1}_{i\notin S}\right)^2 \middle| \ell \notin S\right]$$

$$= \sum_i a_i^2\, \mathbb{P}_s\left(i \notin S \mid \ell \notin S\right) + 2\sum_{i<j} a_i a_j\, \mathbb{P}_s\left(i \notin S, j \notin S \mid \ell \notin S\right).$$

As before, we have to take care of equality cases. Using Lemma D.5 and our assumption that $\sum_i a_i = 0$, we find

$$\mathbb{E}_s\left[H_S^2 \middle| \ell \notin S\right] = \left(\sum_{i\neq\ell} a_i^2\right)\frac{n-s-1}{n-1} + a_\ell^2 + \left(2\sum_{\substack{i<j\\i,j\neq\ell}} a_i a_j\right)\frac{(n-s-1)(n-s-2)}{(n-1)(n-2)}$$

$$+ \left(2a_\ell\sum_{i\neq\ell} a_i\right)\frac{n-1-s}{n-1}$$

$$= \left(\sum_i a_i^2\right)\frac{n-s-1}{n-1} + \left(2\sum_{i<j} a_i a_j\right)\frac{(n-s-1)(n-s-2)}{(n-1)(n-2)} - a_\ell^2\frac{s(n-2s)}{(n-1)(n-2)}$$

$$= \frac{s(n-s-1)}{n-1)(n-2)}\sum_i a_i^2 - \frac{s(n-2s)}{(n-1)(n-2)}a_\ell^2\,,$$

since $2\sum_{i<j} a_i a_j = -\sum_i a_i^2$. Putting everything together, we obtain

$$\mathrm{Var}_s(H_S \mid \ell \notin S) = \frac{ns(n-s-1)}{(n-1)(n-2)}\left[\widehat{\mathrm{Var}}(a) - \frac{1}{n-1}a_\ell^2\right],$$

from which we deduce the result. $\qquad\square$

D.2. Probability computations

Lemma D.4 (Exact expressions). *Let a, b, c be distinct elements of $[n]$. Then*

$$\begin{cases} \mathbb{P}_s\left(a \notin S\right) = \frac{n-s}{n} \\ \mathbb{P}_s\left(a \notin S, b \notin S\right) = \frac{(n-s)(n-1-s)}{n(n-1)} \\ \mathbb{P}_s\left(a \notin S, b \in S\right) = \frac{s(n-s)}{n(n-1)} \\ \mathbb{P}_s\left(a \notin S, b \notin S, c \notin S\right) = \frac{(n-s)(n-s-1)(n-s-2)}{n(n-1)(n-2)} \\ \mathbb{P}_s\left(a \notin S, b \in S, c \notin S\right) = \frac{s(n-s-1)(n-s)}{n(n-1)(n-2)} \end{cases} \tag{37}$$

Proof. Similar to Lemma 4 in Mardaoui & Garreau (2021). $\qquad\square$

Lemma D.5 (Exact expressions, conditional). *Let a, b, ℓ be distinct elements of $[n]$. Then*

$$\begin{cases} \mathbb{P}_s\left(a \notin S \mid \ell \notin S\right) = \frac{n-1-s}{n-1} \\ \mathbb{P}_s\left(a \in S \mid \ell \notin S\right) = \frac{s}{n-1} \\ \mathbb{P}_s\left(a \notin S, b \notin S \mid \ell \notin S\right) = \frac{(n-s-1)(n-s-2)}{(n-1)(n-2)} \end{cases} \tag{38}$$

Proof. Let us prove the first statement, the other ones are similar. By Bayes formula,

$$\mathbb{P}_s\left(a \notin S \mid \ell \notin S\right) = \frac{\mathbb{P}_s\left(a \notin S, \ell \notin S\right)}{\mathbb{P}_s\left(\ell \notin S\right)}.$$

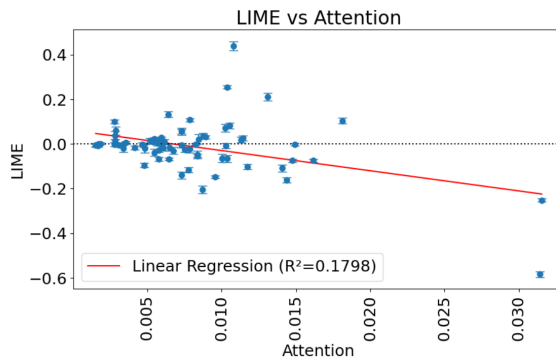We now simply use Lemma D.4. $\qquad\square$

Figure 6. Relation between LIME weights and attention for a 6-layer 6-head attention-based classifier. Attention weights correspond to the average attention over the 6 heads of the first layer. Error bars represent the standard deviation of LIME weights on 5 repetition.

E. Experiments on multi-layer architecture

We have conducted numerical experiments on a multi-head, multi-layer architecture. We trained a classifier with 6 layers and 6 heads on the IMDb dataset (refer to Appendix F), achieving an accuracy of 82.22%. Our interest lies in exploring the relationship between LIME and the attention weights. We measured the correlation between LIME coefficients and the α-avg (refer to Eq. (19)) for the first layer. An illustration is available at Figure 6, where the document corresponds to Figure 4 of our paper. The Pearson's correlation in this case is $\rho = -0.424$. We attribute the negative sign to the document being classified as negative (as in Figure 4). Attention weights consistently fall within the range of $[0, 1]$. Considering the absolute values, the rankings of the two explanations are closely aligned, and the correlation is $\rho = 0.672$. Although we cannot explicitly state the dependency for multi-layers as in Eq. (19), our experiments suggest a significant relationship. We conclude that the attention weights are interconnected with LIME coefficients, which adapt more effectively to the model. We are currently conducting broader experiments and will incorporate their results and subsequent discussions into the manuscript.

F. Experiments

In this section, we report technical details for the model and the experiments. Any of the experiments presented in this paper have been performed on a `PyTorch` implementation of the model presented in Section 2 and ran on one GPU Nvidia A100.

Code. The full code is available at https://github.com/gianluigilopardo/attention_meets_xai.

Model. The model parameters were set as follows: $T_{\max} = 256$, $d_e = 128$, $d_{\text{att}} = 64$, $d_{\text{out}} = 64$.

Dataset. We trained the model on the `IMDB` dataset (Maas et al., 2011), which was preprocessed using standard tokenization and padding techniques. The dataset was split into training, validation, and test sets with sizes of $20,000$, $5,000$, and $25,000$ samples, respectively.

Training. The model was trained for 10 epochs using a batch size of 16. We employed the `AdamW` optimizer with a learning rate of 0.0001 and used cross-entropy loss as the optimization objective.