Beyond Intuition: Rethinking Token Attributions inside Transformers

Anonymous authors
Paper under double-blind review

Abstract

The multi-head attention mechanism, or rather the Transformer-based models have always been under the spotlight, not only in the domain of text processing, but also for computer vision. Several works have recently been proposed around exploring the token attributions along the intrinsic decision process. However, the ambiguity of the expression formulation can lead to an accumulation of error, which makes the interpretation less trustworthy and less applicable to different variants. In this work, we propose a novel method to approximate token contributions inside Transformers. We start from the partial derivative to each token, divide the interpretation process into *attention perception* and *reasoning feedback* with the chain rule and explore each part individually with explicit mathematical derivations. In attention perception, we propose the head-wise and token-wise approximations in order to learn how the tokens interact to form the pooled vector. As for reasoning feedback, we adopt a noise-decreasing strategy by applying the integrated gradients to the last attention map. Our method is further validated qualitatively and quantitatively through the faithfulness evaluations across different settings: single modality (BERT and ViT) and bi-modality (CLIP), different model sizes (ViT-L) and different pooling strategies (ViT-MAE) to demonstrate the broad applicability and clear improvements over existing methods.

1 Introduction

The Transformer (Vaswani et al., 2017) and its variants (Dosovitskiy et al., 2021; Devlin et al., 2019; Brown et al., 2020) take a leading place in the domain of natural language processing (NLP), where texts are firstly tokenized into words or sub-words, identified as tokens, and then fed into the deep neural network. The main architecture of Transformer consists of several attention blocks, where the query-relevant tokens are captured and combined to make a new representation. With its huge success in NLP, more and more Transformer-based models have been proposed for multiple tasks, such as image classification (Dosovitskiy et al., 2021), object detection (Carion et al., 2020), VQA (Antol et al., 2015) and GQA (Hudson & Manning, 2019). For images, tokens are usually the patches which are uniformly segmented (Dosovitskiy et al., 2021), obtained with a shift window (Liu et al., 2021) or captured with an object detector (Tan & Bansal, 2019; Li et al., 2019).

Explaining how tokens are mixed and used inside the Transformer to make the final prediction can help a lot in understanding, debugging and refining the model. Some classic interpretation methods can be applied with modifications considering a Transformer literature, such as the input gradient based methods (Selvaraju et al., 2017; Sundararajan et al., 2017; Smilkov et al., 2017). Regardless of their versatility for all differentiable models, the explanation results can be noisy to certain extent, due to the saturation and vanishing of gradients caused by numerous non-linear components in the deep models. Some attention-based explanation methods have also been proposed for the Transformers (Michel et al., 2019; Abnar & Zuidema, 2020; Chefer et al., 2021a;b; Hao et al., 2021), despite the disputation about the legitimacy of attentions being an explanation (Jain & Wallace, 2019; Wiegreffe & Pinter, 2019; Kobayashi et al., 2020). With more characteristics inside Transformer taken into account, these attention-based algorithms generally provide more faithful explanation results than simply adopting the raw attentions.

However, existing attention-based Transformer explanation methods are built either on too strong assumptions, or without clear theoretical frameworks. Rollout (Abnar & Zuidema, 2020) takes skip connections into consideration but ignores the existing effects of linear projections inside attention blocks (as detailed in Section 3.2). Transformer

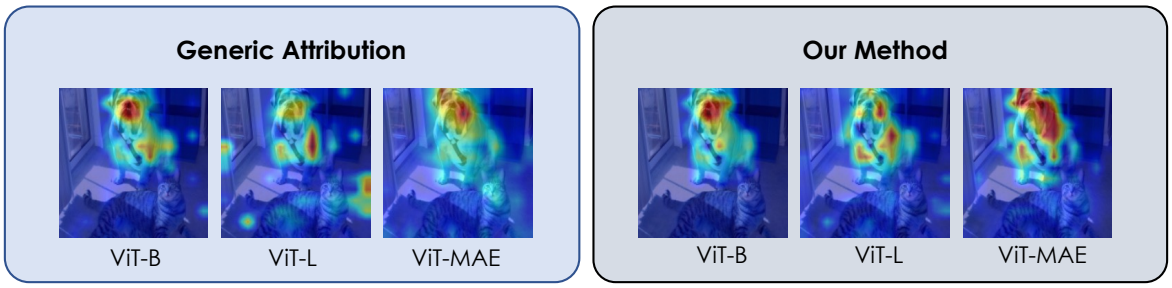

Figure 1: With different ViT variants, we observe an accumulation of error for the intuitive interpretation method Generic Attribution when the model gets deeper or adopts a global pooling strategy, which makes the results contradictory with the model performances.

Attribution (Chefer et al., 2021a) and Generic Attribution (Chefer et al., 2021b) combine the gradients with layer-wise relevance propagation (Binder et al., 2016) (LRP) or attention maps along a rolling out path, and eliminate the negative components in each attention block. Though gradients in a way reflect the direct influence to the final objective, involving them intuitively and masking the negatives in each attention block risk losing useful information from relevance scores or attention maps.

In this work, we intend to bring up the potential risks of introducing the methods intuitively. The unidentified errors from intuitive designs can be disclosed and magnified under different settings, causing the methods with good performances in the vanilla models fail in explaining their variants. We show in Figure 1 some visualizations with Generic Attribution to different types of Vision Transformers (Dosovitskiy et al., 2021; He et al., 2022). In spite of its good performances in explaining ViT-Base, the visualizations on other ViT variants show some contradictions to the models' properties, where the model with better performances (ViT-L) is visualized with a focus on less semantic pixels, and the explanation results do not correspond with the semantics enhancement brought by the masked pretraining technique (ViT-MAE). These contradictions decrease the trustworthiness of interpretation methods and make them less applicable to all Transformer-based models.

Contributions. In order to explicitly explain how the model predicts and theoretically formalize the token interactions inside Transformers, we introduce a novel explanation method with main contributions as following:

1) Based on the partial derivative of the loss function to each token, we propose a novel interpretation framework with the chain rule, dividing the whole process into the *Attention Perception* and *Reasoning Feedback*, as shown in Figure 2. We believe that this framework can inspire more interpretation methods with other approximation strategies.

2) In attention perception, we look into the relation between the input and output in each attention block and mathematically derive two recurrence formulas with our head-wise and token-wise attention maps, where the token-wise one achieves $20\times$ more accurate on average than Attention Rollout. By recurring from the first to the last attention block, we reach the expression indicating the contributions of input tokens in the pooled vector before prediction head.

3) To validate the faithfulness of the proposed explanation algorithm, we follow the previous evaluation methods (Samek et al., 2017; Vu et al., 2019; DeYoung et al., 2020), and compare ours with existing algorithms. The evaluation settings cover different modalities, different sizes and different pooling strategies concerning Transformers, including BERT-Base (Devlin et al., 2019) for texts, ViT-Base (Dosovitskiy et al., 2021) for images, CLIP (Radford et al., 2021) for bi-modality, ViT-Large (Dosovitskiy et al., 2021) for the large model, and ViT-MAE (He et al., 2022) for the global pooling strategy. Our approach outperforms other strong baselines (e.g., (Abnar & Zuidema, 2020; Chefer et al., 2021a;b)) through quantitative metrics and qualitative visualizations, and shows better applicability to various settings. More analyses and ablation studies are also provided.

2 Related work

Transformer-based Models. For the first time being proposed by Vaswani et al. (2017), the Transformer has been rapidly and widely used for almost all kinds of deep learning tasks. A variety of derived methods (Dosovitskiy et al., 2021; Devlin et al., 2019; Brown et al., 2020; Liu et al., 2019) has sprung up in recent years. These Transformer-based

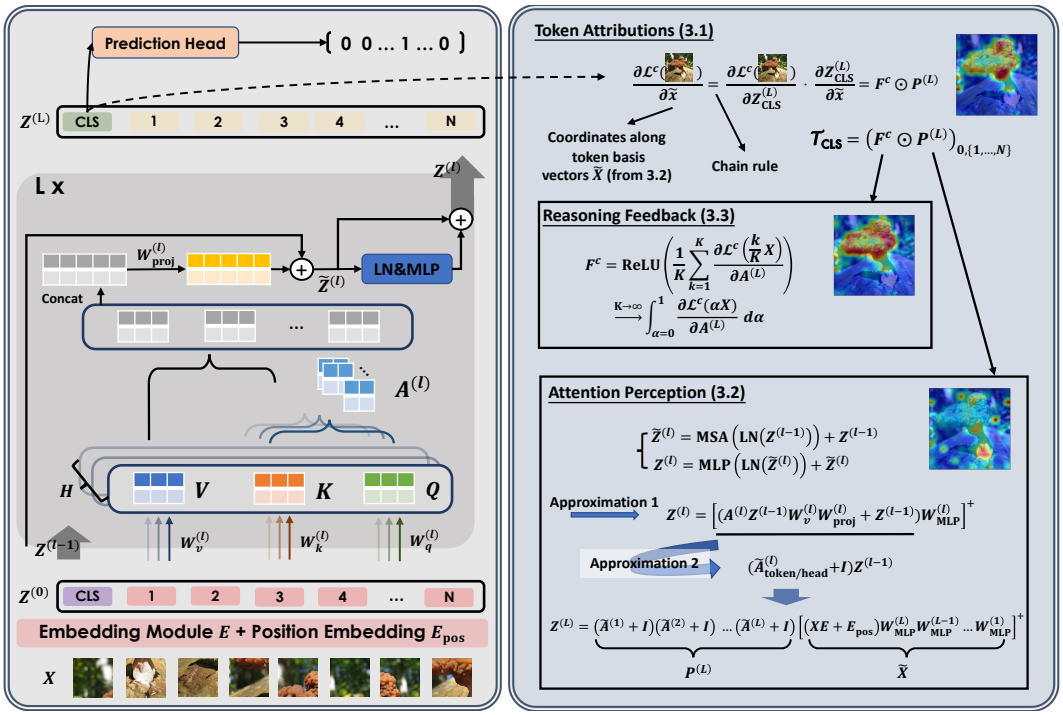

Figure 2: Illustration of our method with ViT (Dosovitskiy et al., 2021). We show in the *left* how information propagates inside ViT. For our method in the *right*, we start from the partial derivative, divide the whole process into *Attention Perception* and *Reasoning Feedback* with the chain rule, and analyze each sub-process to obtain the final expression.

models have become the state of the art in most of the NLP benchmarks and stepped in the domain of computer vision as well (Dosovitskiy et al., 2021; Liu et al., 2021). Vision Transformer (ViT) (Dosovitskiy et al., 2021) takes a sequence of image patches, regraded as tokens, into successive attention blocks and feeds the last class token [CLS] into prediction head for downstream tasks. With its broad usage covering both image and text, the Transformer architecture has been explored for a lot of bi-modal models, e.g., VisualBERT (Li et al., 2019), LXMERT (Tan & Bansal, 2019), CLIP (Radford et al., 2021) and VLM (Xu et al., 2021). A single or dual self-attention/cross-attention based encoder is used to get the representations of both text and image. Thus, being capable to explain the decision process of these Transformer-based models is rather important for a deeper understanding and a refinement for better performance.

Explainability of Deep Models. Deep models are well known for their superb performance as well as their black-box nature. Before Transformers, many explanation methods have been proposed to explain the deep models. For instance, as a post-hoc local explanation method, LIME (Ribeiro et al., 2016) locally explains the deep model at a single data point. Many methods based on gradients have also been proposed because of the differentiability of most deep models. Smooth Gradient (SG) (Smilkov et al., 2017) and Integrated Gradient (IG) (Sundararajan et al., 2017) are two explanation algorithms to produce input gradient based saliency maps, through adding noisy inputs, or using the integral respectively. More methods focus on specific model structures, e.g., CAM/Grad-CAM (Zhou et al., 2016; Selvaraju et al., 2017; Wang et al., 2020) for convolutional networks, GNNExplainer (Ying et al., 2019) for graph neural networks, and GAN dissection (Bau et al., 2019) for generative adversarial networks.

Explainability for Transformers. Some Transformer-specific explanation methods have also been proposed recently, especially towards the goal of better understanding the characteristic components inside. Previous studies (Bahdanau et al., 2015; Xu et al., 2015; Choi et al., 2016) show that raw attention maps can be used as explanations for attention-based models. Considering the multi-head mechanism, Michel et al. (2019) find that the contributions of heads vary a lot from one to another, and therefore propose that pruning the unimportant heads has little impact to the model. Attention Rollout (Abnar & Zuidema, 2020) assumes that the input tokens are linearly combined based on the attention weights and takes the [CLS] token as explanations. Generic Attribution (Chefer et al., 2021b) generalizes the idea of Rollout and adds the gradient information to each attention map, while Transformer Attribution (Chefer et al., 2021a) exploits LRP (Binder et al., 2016) and gradients together for getting the explanations.

3 Method

This section introduces the explanation expression we propose for Transformers, with a roadmap presented in Figure 2 to illustrate briefly our ideas. To be more specific, we start by recalling the concept of Multi-Head Self-Attention (MSA) blocks and introducing the setup of our problem. We then explain in Section 3.1 how our explanation expression formulates by dividing the whole process into *Attention Perception* and *Reasoning Feedback* based on the chain rule and a change of basis. Section 3.2 analyzes how the tokens interact before the prediction head, identifies the basis vectors and derives the expression of *Attention Perception* explicitly with our head-wise and token-wise approximations. Section 3.3 introduces the technique of integration we adopt to decrease the noise of *Reasoning Feedback* in the derived expression. We further provide in the appendix a table of notations to help the reading.

Preliminaries. We assume a Transformer-based model consisting of L blocks. The input of the model is a sequence of N tokens. After the embedding module $\mathbf{E}$, the tokens are turned into vectors of dimension D, with a special [CLS] token at the top and a positional embedding $\mathbf{E}_{\text{pos}} = [\mathbf{E}_{\text{pos},0}; \mathbf{E}_{\text{pos},1}; \ldots; \mathbf{E}_{\text{pos,N}}]$. Let l be the l-th attention block composed of layer normalisation, multi-head self-attention, skip connections and projection layers. We denote the input of the block l as $\mathbf{Z}^{(l-1)}$, and the output $\mathbf{Z}^{(l)}$. In this MSA block, the attentions $\mathbf{A}^{(l)}$, defined as $\mathrm{softmax}(\mathrm{Q} \cdot \mathrm{K}^{\mathrm{T}}/\sqrt{\mathrm{d_h}})$, capture the important tokens from a certain query and combines them to make an output. Here for self-attention, query (Q), key (K) and value (V) are from trainable transformations of the same input data. Layer Normalisation (LN) (Ba et al., 2016) re-scales the input for a single training case. The MLP module consists of two linear layers with an activation function ReLU/GeLU in between. In brief, we have

$$\mathbf{Z}^{(0)} = [X_{\text{CLS}}; X_1\mathbf{E}; X_2\mathbf{E}; ...; X_N\mathbf{E}] + \mathbf{E}_{\text{pos}}, \qquad \mathbf{E} \in \mathbb{R}^{(d_X)\times D}, \mathbf{E}_{\text{pos}} \in \mathbb{R}^{(N+1)\times D} \quad , \tag{1}$$

$$\tilde{\mathbf{Z}}^{(l)} = \mathrm{MSA}(\mathrm{LN}(\mathbf{Z}^{(l-1)})) + \mathbf{Z}^{(l-1)}, \qquad l = 1, ..., L \quad , \tag{2}$$

$$\mathbf{Z}^{(l)} = \mathrm{MLP}(\mathrm{LN}(\tilde{\mathbf{Z}}^{(l)})) + \tilde{\mathbf{Z}}^{(l)}, \qquad l = 1, ..., L \quad . \tag{3}$$

Note that the MSA module operates on the sub-spaces of dimension D_h, where h refers to the head index, and we have H heads so that $HD_h = D$.

Most Transformers perform the classification task via the last [CLS] embedding, while some use global pooling instead (He et al., 2022). In this section, we introduce our method particularly with [CLS] pooling, but the same methodology can also be applied for other cases.

3.1 Token Attributions inside Transformers

Let X be the input and $\mathcal{L}^c$ the loss function while predicting class c. The aim of our method is to obtain the importance of each input token with respect to the decision process. Using the back-propagated gradients to the input $\frac{\partial \mathcal{L}^c(X)}{\partial X}$ could be an answer, but here we would like to study directly the entire attribution of each token, since the transformation from the obtained high-dimensional explanation results to a single scalar which represents the contribution of each token can bring unexpected noise. By formulating the input-output relations in attention blocks (details in Section 3.2), we propose a set $\mathbb{B}$ of vectors,

$$\mathbb{B} = \{\tilde{X}_{\text{CLS}}, \tilde{X}_1, \ldots, \tilde{X}_N\} \quad , \tag{4}$$

where each element represents the corresponding token. These basis vectors will be identified later in Eq. (12). In this way, we can rewrite the input vector X as $X_{\mathbb{B}} = (\tilde{x}_0, \tilde{x}_1, \ldots, \tilde{x}_N)$, where $\tilde{x}_0$ is the coordinate over $\tilde{X}_{\text{CLS}}$ and $\tilde{x}_i$ is the coordinate over $\tilde{X}_i$ for $i \in \{1, \ldots, N\}$. Thus, we can define the attribution of the i-th token $\frac{\partial \mathcal{L}^c(X)}{\partial \tilde{x}_i}$.

From the chain rule, we develop $\frac{\partial \mathcal{L}^c(X)}{\partial \tilde{x}_i}$ and denote $\mathbf{Z}^{(L)}_{\text{CLS}}$ the [CLS] token of the output in the last L-th attention block, having

$$\frac{\partial \mathcal{L}^c(X)}{\partial \tilde{x}_i} = \frac{\partial \mathcal{L}^c(X)}{\partial \mathbf{Z}^{(L)}_{\text{CLS}}} \cdot \frac{\partial \mathbf{Z}^{(L)}_{\text{CLS}}}{\partial \tilde{x}_i} = \frac{\partial \mathcal{L}^c(X)}{\partial (\mathbf{Z}^{(L)}_{\text{CLS}})_{\mathbb{B}}} \cdot \frac{\partial (\mathbf{Z}^{(L)}_{\text{CLS}})_{\mathbb{B}}}{\partial \tilde{x}_i} \quad . \tag{5}$$

With the analyses in Approximation 2, $\mathbf{Z}^{(L)}_{\text{CLS}}$ can be rewritten under the basis $\mathbb{B}$ as $(\mathbf{Z}^{(L)}_{\text{CLS}})_{\mathbb{B}} = \mathbf{P}^{(L)}_{0,\cdot}$, indicating the attributions of each input token to the last [CLS] before the prediction head. Here we make an approximation

that $\frac{\partial \mathcal{L}^c(X)}{\partial \mathbf{P}^{(L)}_{0,\cdot}} \approx \frac{\partial \mathcal{L}^c(X)}{\partial \mathbf{A}^{(L)}_{0,\cdot}}$, which could be tenable for the cases where raw attentions can in a way represent the attributions in last [CLS] token. So we have $\frac{\partial \mathcal{L}^c(X)}{\partial (\mathbf{Z}^{(L)}_{\text{CLS}})_{\mathbb{B}}} \approx \frac{\partial \mathcal{L}^c(X)}{\partial \mathbf{A}^{(L)}_{0,\cdot}}$. As for the second term $\frac{\partial (\mathbf{Z}^{(L)}_{\text{CLS}})_{\mathbb{B}}}{\partial \tilde{x}_i}$, we can have $\frac{\partial (\mathbf{Z}^{(L)}_{\text{CLS}})_{\mathbb{B}}}{\partial \tilde{x}_i} = (0, \ldots, \mathbf{P}^{(L)}_{0,i}, \ldots, 0)$.

Now plugging the above formulas into Eq. (5), we can compute the attribution of i-th token, which is

$$\frac{\partial \mathcal{L}^c(X)}{\partial \tilde{x}_i} \approx \frac{\partial \mathcal{L}^c(X)}{\partial \mathbf{A}^{(L)}_{0,\cdot}} \cdot \frac{\partial (\mathbf{Z}^{(L)}_{\text{CLS}})_{\mathbb{B}}}{\partial \tilde{x}_i} = \frac{\partial \mathcal{L}^c(X)}{\partial \mathbf{A}^{(L)}_{0,i}} \mathbf{P}^{(L)}_{0,i} \quad . \tag{6}$$

Combining all the tokens together and using $\mathbf{F}^c$ to denote $\frac{\partial \mathcal{L}^c(X)}{\partial \mathbf{A}^{(L)}}$, we reach the expression of our explanation method:

$$\left(\frac{\partial \mathcal{L}^c(X)}{\partial \tilde{x}_1}, \ldots, \frac{\partial \mathcal{L}^c(X)}{\partial \tilde{x}_i}, \ldots, \frac{\partial \mathcal{L}^c(X)}{\partial \tilde{x}_N} \right) \approx \left(\mathbf{F}^c \odot \mathbf{P}^{(L)} \right)_{0,\{1 \ldots N\}} \quad . \tag{7}$$

Our Explanation Method. *For the Transformer-based model, the attributions of partitioned tokens to the final decision $\mathcal{T}$ can be expressed as a Hadamard product of two parts:*

$$\mathcal{T} = \left(\mathbf{P}^{(L)} \odot \mathbf{F}^c \right)_{\mathcal{P}} \quad . \tag{8}$$

We denote the first part $\mathbf{P}^{(L)}$, representing how the input tokens integrate in L successive attention blocks and being regarded as a process of representation learning, so called *attention perception*. The second part, denoted as $\mathbf{F}^c$, implies how the last [CLS] token is used for predicting a certain class c, so called *reasoning feedback*. Their details will be introduced in Sections 3.2 and 3.3 respectively. The subscript $\mathcal{P} \in \{\text{cls}, \text{glp}, \ldots\}$ is determined by the pooling strategy, such as [CLS] pooling and global pooling. In [CLS] pooling, it refers to the indices $\{(0,1), \ldots, (0,N)\}$ of the matrix $\mathcal{T}_{\text{cls}} = (\mathbf{P}^{(L)} \odot \mathbf{F}^c)_{0,\{1 \ldots N\}}$. In global pooling, by replacing $\mathbf{Z}^{(L)}_{\text{CLS}}$ with $[\mathbf{Z}^{(L)}_1; \ldots; \mathbf{Z}^{(L)}_N]$, we obtain its expression $\mathcal{T}_{\text{glp}} = \sum_{k=1}^{k=N} (\mathbf{P}^{(L)} \odot \mathbf{F}^c)_{k,\{1 \ldots N\}}$.

3.2 Attention Perception

In this subsection, in order to identify $\mathbf{P}^{(L)}$, we analyze how the input tokens interact to obtain the last [CLS] representation. We first unfold each attention block to derive the relation between the input and output of the block.

Approximation 1. *In MSA blocks with attentions $\mathbf{A}^{(l)}$, the relations between input $\mathbf{Z}^{(l-1)}$ and output $\mathbf{Z}^{(l)}$ can be approximated in the form of $\mathbf{Z}^{(l)} \approx [(\mathbf{A}^{(l)} \mathbf{Z}^{(l-1)} W^{(l)} + \mathbf{Z}^{(l-1)}) W^{(l)}_{\text{MLP}}]^+$.*

Justification. The non-linear components such as layer normalisation (LN) and activation function play an important role in deep neural networks. Here we neglect the LN's effect, because it operates on individual tokens and does not make tokens interacted. And we consider the result of ReLU/GeLU as an elimination of the negative feature maps. Thus we simplify the information flow in the attention block as linear combinations and transformations for all the tokens with a removal of all the negative terms at the output:

$$\tilde{\mathbf{Z}}^{(l)} \underset{\text{assum.}}{\overset{\text{linear}}{=}} \mathbf{A}^{(l)} \cdot (\mathbf{Z}^{(l-1)} W^{(l)}_v) W^{(l)}_{proj} + b^{(l)}_{proj} + \mathbf{Z}^{(l-1)} \quad , \qquad l = 1, ..., L \quad , \tag{9}$$

$$\mathbf{Z}^{(l)} \underset{\text{assum.}}{\overset{\text{linear}}{=}} ((\tilde{\mathbf{Z}}^{(l)} W^{(l)}_{\text{MLP},1} + b^{(l)}_{\text{MLP},1}) W^{(l)}_{\text{MLP},2} + b^{(l)}_{\text{MLP},2} + \tilde{\mathbf{Z}}^{(l)})^+ \quad , \qquad l = 1, ..., L \quad , \tag{10}$$

where $W^{(l)}_v, W^{(l)}_{proj} \in \mathbb{R}^{D \times D}$ and $b^{(l)}_{proj} \in \mathbb{R}^D$ indicate the linear transformations inside Eq. (2), and $b^{(l)}_{\text{MLP},1} \in \mathbb{R}^{D_{\text{hidden}}}, b^{(l)}_{\text{MLP},2} \in \mathbb{R}^D$, $W^{(l)}_{\text{MLP},1} \in \mathbb{R}^{D \times D_{\text{hidden}}}$, $W^{(l)}_{\text{MLP},2} \in \mathbb{R}^{D_{\text{hidden}} \times D}$ in Eq. (3).

After putting Eq. (9) into Eq. (10), we rearrange the formula and obtain the relations between the input and output of the attention block l as:

$$\mathbf{Z}^{(l)} = [(\mathbf{A}^{(l)} \mathbf{Z}^{(l-1)} W^{(l)} + \mathbf{Z}^{(l-1)}) W^{(l)}_{\text{MLP}} + B^{(l)}]^+ \quad , \tag{11}$$

where $W_{\text{MLP}}^{(l)} = W_{\text{MLP},1}^{(l)} W_{\text{MLP},2}^{(l)} + I$, $W^{(l)} = W_v^{(l)} W_{proj}^{(l)}$ and $B^{(l)} = b_{proj}^{(l)}(W_{\text{MLP},1}^{(l)} W_{\text{MLP},2}^{(l)} + I) + b_{\text{MLP},1}^{(l)} W_{\text{MLP},2}^{(l)} + b_{\text{MLP},2}^{(l)}$. We will neglect $B^{(l)}$ because it is shared by all tokens and does not make tokens interacted. Thus, we reach the Approximation 1.

Approximation 2. *If we can find* $\tilde{\mathbf{A}}^{(l)}$ *satisfying* $\mathbf{A}^{(l)}\mathbf{Z}^{(l-1)}W^{(l)} + \mathbf{Z}^{(l-1)} \approx (\tilde{\mathbf{A}}^{(l)} + I)\mathbf{Z}^{(l-1)}$ *, the relations between the input tokens and the output vectors of last attention block* $\mathbf{Z}^{(L)} = (\mathbf{Z}_{\text{CLS}}^{(L)}, \mathbf{Z}_1^{(L)}, \ldots, \mathbf{Z}_N^{(L)})^{\text{T}}$ *can be expressed as* $\mathbf{Z}^{(L)} \approx \mathbf{P}^{(L)}(\tilde{X}_{\text{CLS}}, \tilde{X}_1, \ldots, \tilde{X}_N)^{\text{T}}$ *, where* $\mathbf{P}^{(L)} = (\tilde{\mathbf{A}}^{(L)} + I)\ldots(\tilde{\mathbf{A}}^{(1)} + I)$ *and* $\tilde{X}_i = (X_i\mathbf{E} + \mathbf{E}_{\text{pos,i}})W_{\text{MLP}}^{(L)} \ldots W_{\text{MLP}}^{(1)}$.

Justification. With $\mathbf{A}^{(l)}\mathbf{Z}^{(l-1)}W^{(l)} + \mathbf{Z}^{(l-1)} \approx (\tilde{\mathbf{A}}^{(l)} + I)\mathbf{Z}^{(l-1)}$, we can obtain the recurrence relation $\mathbf{Z}^{(l)} = [(\tilde{\mathbf{A}}^{(l)} + I)\mathbf{Z}^{(l-1)}W_{\text{MLP}}^{(l)}]^+$. Hence, the conclusion can be reached directly from recursion. We recur from the last block L to the first block and obtain the relation between output of block L and input tokens:

$$\begin{pmatrix} \mathbf{Z}_{\text{CLS}}^{(L)} \\ \mathbf{Z}_1^{(L)} \\ \ldots \\ \mathbf{Z}_N^{(L)} \end{pmatrix} \approx (\tilde{\mathbf{A}}^{(L)} + I)\ldots(\tilde{\mathbf{A}}^{(1)} + I) \begin{pmatrix} (X_{\text{CLS}} + \mathbf{E}_{\text{pos,0}})\tilde{\mathbf{W}} \\ (X_1\mathbf{E} + \mathbf{E}_{\text{pos,1}})\tilde{\mathbf{W}} \\ \ldots \\ (X_N\mathbf{E} + \mathbf{E}_{\text{pos,N}})\tilde{\mathbf{W}} \end{pmatrix}^+ = \mathbf{P}^{(L)} \begin{pmatrix} \tilde{X}_{\text{CLS}} \\ \tilde{X}_1 \\ \ldots \\ \tilde{X}_N \end{pmatrix} \quad , \tag{12}$$

where $\tilde{\mathbf{W}} = W_{\text{MLP}}^{(L)} \ldots W_{\text{MLP}}^{(1)}$ and $\tilde{X}_i = (X_i\mathbf{E} + \mathbf{E}_{\text{pos,i}})\tilde{\mathbf{W}}$. With the normalization components in neural networks and the assumption of independence between the input tokens, $\{\tilde{X}_{\text{CLS}}, \tilde{X}_1 \ldots, \tilde{X}_N\}$ forms a set of basis vectors.

Remark. The recurrence relation of Attention Rollout method (Abnar & Zuidema, 2020) can be rediscovered under the assumption that $W^{(l)} = I$. With this assumption, we have $\mathbf{A}^{(l)}\mathbf{Z}^{(l-1)}W^{(l)} + \mathbf{Z}^{(l-1)} = \mathbf{A}^{(l)}\mathbf{Z}^{(l-1)} + \mathbf{Z}^{(l-1)}$ so that $\tilde{\mathbf{A}}^{(l)} = \mathbf{A}^{(l)} + I$. Hence, we obtain the expression of Attention Rollout with the conclusions from Approximation 2: $\mathbf{Rollout} = (\mathbf{A}^{(L)} + I)\ldots(\mathbf{A}^{(1)} + I)$.

However, the assumption of regrading it as an identity matrix is too strong. Although $W^{(l)}$ only operates on the embedding dimension, due to $\mathbf{A}^{(l)}$, it also contributes to a token interaction. In order to study its effect and find a more accurate $\tilde{\mathbf{A}}^{(l)}$ expression, we derive $\mathbf{O}^{(l)} = \mathbf{A}^{(l)}\mathbf{Z}^{(l-1)}W^{(l)}$ by two approximations: $\tilde{\mathbf{A}}_{\text{token}}^{(l)}$ and $\tilde{\mathbf{A}}_{\text{head}}^{(l)}$. We name the token-wise attention map using "token" since it contributes the effect of $W^{(l)}$ to $\mathbf{A}^{(l)}$ as a norm difference between token vectors, and name the head-wise one for a difference between heads.

Token-wise Attention Map. By the definition of matrix product, we develop $\mathbf{O} = \mathbf{AZ}W$ and have

$$\mathbf{O}_{i,j} = \sum_m \mathbf{A}_{i,m}(\sum_k \frac{\mathbf{Z}_{m,k}W_{k,j}}{\mathbf{Z}_{m,j}})\mathbf{Z}_{m,j} \underset{\text{wise}}{\overset{\text{token}}{\approx}} \sum_m (\mathbf{A}_{i,m}\frac{\|(\mathbf{Z}W)_{m,\cdot}\|}{\|\mathbf{Z}_{m,\cdot}\|})\mathbf{Z}_{m,j} \tag{13}$$

$$\Longrightarrow \quad \mathbf{O}^{(l)} = \mathbf{A}^{(l)}\mathbf{Z}^{(l-1)}W^{(l)} \underset{\text{wise}}{\overset{\text{token}}{\approx}} \tilde{\mathbf{A}}_{\text{token}}^{(l)}\mathbf{Z}^{(l-1)} \quad , \tag{14}$$

where $(\tilde{\mathbf{A}}_{\text{token}}^{(l)})_{\cdot,i} = \alpha_i \mathbf{A}_{\cdot,i}^{(l)}$ and $\alpha_i = \frac{\|(\mathbf{Z}^{(l-1)}W^{(l)})_{i,\cdot}\|}{\|\mathbf{Z}_{i,\cdot}^{(l-1)}\|}$. So with $\tilde{\mathbf{A}}_{\text{token}}^{(l)}$, we obtain $\mathbf{Z}^{(l)} \underset{\text{wise}}{\overset{\text{token}}{\approx}} [(\tilde{\mathbf{A}}_{\text{token}}^{(l)} + I)\mathbf{Z}^{(l-1)}W_{\text{MLP}}^{(l)}]^+$. More derivation details can be found in the appendix.

Head-wise Attention Map. We redevelop $\mathbf{O} = \mathbf{AZ}W$ by taking different heads into consideration, and have

$$\mathbf{O}_{i,j} = \sum_h \sum_m \mathbf{A}_{i,m}^h(\sum_k \frac{\mathbf{Z}_{m,k}W_{k,j}^h}{\mathbf{Z}_{m,j}})\mathbf{Z}_{m,j} \underset{\text{wise}}{\overset{\text{head}}{\approx}} \sum_h \theta_h(\mathbf{A}^h\mathbf{Z})_{i,j} \quad , \tag{15}$$

where $\mathbf{A}^h$ is the attention of head h and θ_h can be regarded as a head weight. Recent works (Michel et al., 2019; Hao et al., 2021) propose a head importance concept I_h by considering the sensitivity of head mask, which means that if I_h is small, removing this head won't affect the model output a lot. A head importance expression is also provided in their work, with the inner product of the attention and its gradient: $I_h = (\mathbf{A}^h)^T \frac{\partial \mathcal{L}^c}{\partial \mathbf{A}^h}$. We adopt here the concept of I_h with normalisation $\theta_h = \frac{I_h}{\sum_h I_h}$ to describe the head weight. Hence, we propose another head-wise type, which is easier to apply in practice, $\tilde{\mathbf{A}}_{\text{head}}^{(l)} = \sum_{h=1}^H (\frac{I_h^{(l)}}{\sum_h I_h^{(l)}})\mathbf{A}_h^{(l)}$ and $\mathbf{Z}^{(l)} \underset{\text{wise}}{\overset{\text{head}}{\approx}} [(\tilde{\mathbf{A}}_{\text{head}}^{(l)} + I)\mathbf{Z}^{(l-1)}W_{\text{MLP}}^{(l)}]^+$.

With the above defined recurrence relations and Approximation 2, we obtain the tokens' interactions $\mathbf{P}^{(L)}$ until the last attention block L:

$$\mathbf{P}^{(L)} = (\tilde{\mathbf{A}}^{(L)} + I) \cdot \mathbf{P}^{(L-1)} = (\tilde{\mathbf{A}}^{(L)} + I) \dots (\tilde{\mathbf{A}}^{(1)} + I) \quad , \tag{16}$$

where $\tilde{\mathbf{A}}^{(l)}$ can be $\tilde{\mathbf{A}}^{(l)}_{\text{token}}$ from Eq. (14) or $\tilde{\mathbf{A}}^{(l)}_{\text{head}}$ from Eq. (15).

Our token-wise attention perception achieves $20\times$ more accurate than Attention Rollout. In order to learn the errors between different approximation strategies and the true value, we conduct several tests for Attention Rollout, token-wise and head-wise attention maps. In each block l, we compare $\mathbf{Z}^{(l)}_{\text{rollout}}$, $\mathbf{Z}^{(l)}_{\text{token}}$ and $\mathbf{Z}^{(l)}_{\text{head}}$ by $\mathbf{Z}^{(l)}_{\text{approx}} = [(\tilde{\mathbf{A}}^{(l)} + I)\mathbf{Z}^{(l-1)} W^{(l)}_{\text{MLP}}]^+$ with the true output $\mathbf{Z}^{(l)}_{\text{true}}$ and calculate the square error $\mathbf{E} = \|\mathbf{Z}^{(l)}_{\text{true}} - \mathbf{Z}^{(l)}_{\text{approx}}\|_2^2$ for each method.

Randomly selected 5000 images from ImageNet val set are used for the test. Both of the averages for our token-wise and head-wise methods are below Attention Rollout, especially the token-wise approximation achieves $20\times$ more accurate than Attention Rollout. Figure 3 presents the errors in different blocks, where we can see that our token-wise method has a more stable performance when the layer gets deeper.

3.3 Reasoning Feedback

Attention perception, taken for the first sub-process, is explained by $\mathbf{P}^{(L)}$ as introduced previously. It remains to the reasoning part $\mathbf{F}^c$, which refers to the stage of making the final decision based on the representations before the prediction head. We explain it by $\frac{\partial \mathcal{L}^c(X)}{\partial \mathbf{A}^{(L)}}$, the back-propagated gradients of the final prediction with respect to the attention map of the last block.

Integrated Gradient for Token-level Importance. Directly calculating the gradient of the last attention map may bring some noise and irrelevant features to the saliency map, due to the lack of sensitivity, which causes gradients to focus on the irrelevant features practically (Sundararajan et al., 2017; Smilkov et al., 2017). In order to reduce the gradient self-induced noise, we use an Integrated Gradient method (IG) (Sundararajan et al., 2017) to get the relevant gradients of the last attention map. We set the baseline to all zero, i.e. a black image or all padding tokens, and a linear path to the input. With Riemann approximation, we discretize the integration, having

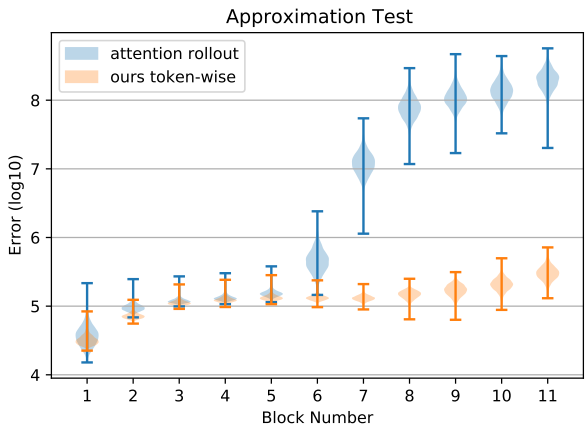

Figure 3: Results of our approximation tests. We notice that our token-wise approximation achieves 10∼100x more accurate than Attention Rollout in different blocks. Besides, the variance of errors in each block also decreases in our token-wise method.

$$\mathbf{F}^c = \text{ReLU}(\frac{1}{K}\sum_{k=1}^{K} \frac{\partial \mathcal{L}^c(\frac{k}{K}X)}{\partial \mathbf{A}^{(L)}}) \overset{\text{K}\to\infty}{\to} \text{ReLU}(\int_{\alpha=0}^{1} \frac{\partial \mathcal{L}^c(\alpha X)}{\partial \mathbf{A}^{(L)}} d\alpha) \quad , \tag{17}$$

where only the positive part of the gradient is taken into consideration and K the total steps.

To make a summary, in this work, we consider the token attributions inside Transformers as the derivative of the loss function $\mathcal{L}^c$ to the input under a novel basis $\mathbb{B} = \{\tilde{X}_{\text{CLS}}, \tilde{X}_1, \dots, \tilde{X}_N\}$. These basis vectors, where each represents certain token, are identified in Eq. (12) from Section 3.2. Based on the chain rule, we divide the derivatives into two parts: *Attention Perception* $\mathbf{P}^{(L)}$ and *Reasoning Feedback* $\mathbf{F}^c$. The first part $\mathbf{P}^{(L)}$ retrospects how the tokens mix up in the successive L blocks. To obtain its expression, we derive a recursive relation inside each block with certain tenable assumptions, i.e. $\mathbf{Z}^{(l)} \approx [(\tilde{\mathbf{A}}^{(l)} + I)\mathbf{Z}^{(l-1)} W^{(l)}_{\text{MLP}}]^+$, where $\tilde{\mathbf{A}}^{(l)}$ denotes our head-wise ($\tilde{\mathbf{A}}^{(l)}_{\text{head}}$) and token-wise ($\tilde{\mathbf{A}}^{(l)}_{\text{token}}$) attention maps taking into account the effect of linear projections. For the second part $\mathbf{F}^c$, explaining the

process of passing the pooled vector for the final prediction, we introduce the technique of integration to decrease the possible noise on irrelevant tokens.

Combining the results of *Attention Perception* and *Reasoning Feedback*, our explanation method $\mathcal{T}_{\text{cls}}$ can be finally expressed as

$$\mathcal{T}_{\text{cls}} = \left(\mathbf{P}^{(L)} \odot \mathbf{F}^{c}\right)_{0,\{1\ldots N\}} = \left(\mathbf{P}^{(L)} \odot \text{ReLU}(\frac{1}{K}\sum_{k=1}^{K}\frac{\partial \mathcal{L}^{c}(\frac{k}{K}X)}{\partial \mathbf{A}^{(L)}})\right)_{0,\{1\ldots N\}}, \tag{18}$$

where $\mathbf{P}^{(L)} = (\tilde{\mathbf{A}}^{(1)} + I)(\tilde{\mathbf{A}}^{(2)} + I)...(\tilde{\mathbf{A}}^{(L)} + I)$ and two types of $\tilde{\mathbf{A}}^{(l)}$ are proposed: token-wise $\tilde{\mathbf{A}}^{(l)}_{\text{token}}$ in Eq. (14) and head-wise $\tilde{\mathbf{A}}^{(l)}_{\text{head}}$ in Eq. (15).

4 Experiments

We validate our proposed explanation method by comparing the results with several strong baselines. The experiment settings are based on two aspects: different modalities and different model versions. The experimental results show the clear advantages and wide applicability of our methods over the others in explaining Transformers.

4.1 Experimental Settings

Faithfulness Evaluation. Following previous works (Abnar & Zuidema, 2020; Chefer et al., 2021a;b; Samek et al., 2017; Vu et al., 2019; DeYoung et al., 2020), we prepare three types of tests for the trustworthiness evaluation:

1) **Perturbation Tests** gradually mask out the tokens of input according to the explanation results and measure the mean accuracy of all these masked inputs, i.e., area under the curve (AUC). There are two kinds of perturbation tests, positive and negative. In the positive perturbation test, tokens are masked from the highest (most relevant) to the lowest. A steep decrease (lower AUC) indicates that the relevance does reflect their importance to the classification score. And vice versa in the negative perturbation test. With the class-specificity for some methods, we adopt their explanation results on two classes: **Predicted**, which refers to the class with highest probability score, and **Target**, which refers to the target class.

2) **Segmentation Tests** equal the explanation to a semantic segmentation of image and compare it with the ground-truth in the dataset. We use here the predicted class to obtain the explanation results for class-specific methods. Each explanation can be evaluated by four metrics: pixel accuracy, mean Intersection over Union (mIoU), mean Average Precision (mAP) and mean F1 (mF1). Both pixel accuracy and mIoU are obtained after binarizing with a threshold set as the average of attribution scores. The metrics mAP and mF1 are calculated by averaging the corresponding scores at each threshold point.

3) **Language Reasoning** comes from a NLP benchmark ERASER (DeYoung et al., 2020) for rationales extraction, whose goal is to extract the input text spans that support the classification. Similar to the segmentation test, with the provided ground-truth, we measure the F1 score of the top-k tokens according to the relevance map, where we consider them to be part of the "rationale".

Baselines and Ours. The following six explanation methods are used as the baselines for a comparison in our experiments: Raw Attention Map (**RAM**), Attention Rollout (Abnar & Zuidema, 2020) (**Rollout**), Generic Attribution (Chefer et al., 2021b) (**GA**), GradCAM (Selvaraju et al., 2017) (**CAM**), Partial LRP (Voita et al., 2019) (**PLRP**) and Transformer Attribution (Chefer et al., 2021a) (**TA**). The appendix provides a detailed description of each method. We choose these baselines according to their explainability literature and applicability to the faithfulness evaluations. Black-box methods, which are computationally expensive and intrinsically different, are not considered here. As for our methods, we have, for each experiment, the token-wise (**Ours-T**) and the head-wise (**Ours-H**) types.

4.2 Transformers on Different Modalities

In this subsection, we focus on the explanation results of all methods with different modalities. We test them on three types of Transformers: **BERT** (Devlin et al., 2019) for texts, **ViT** (Dosovitskiy et al., 2021) for images and

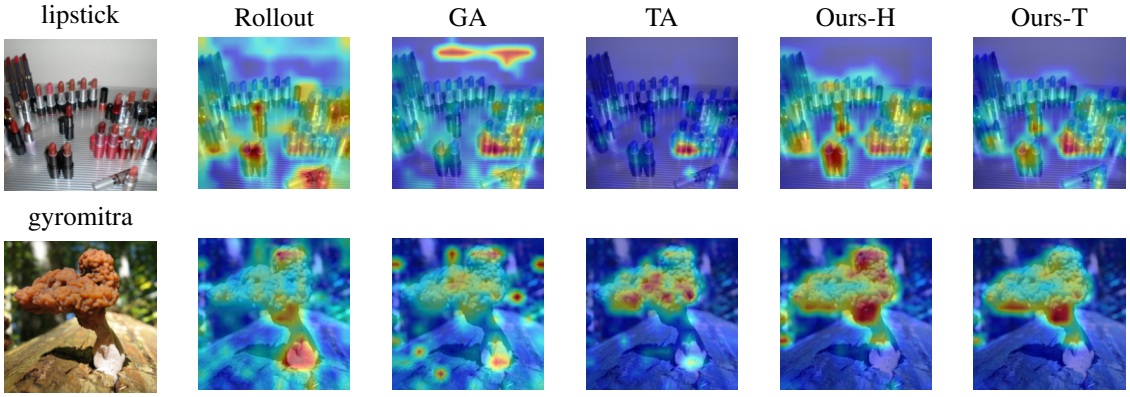

Figure 4: Localization of fine-grained regions for single class prediction of ViT-Base. Both samples are correctly predicted with a high probability score.

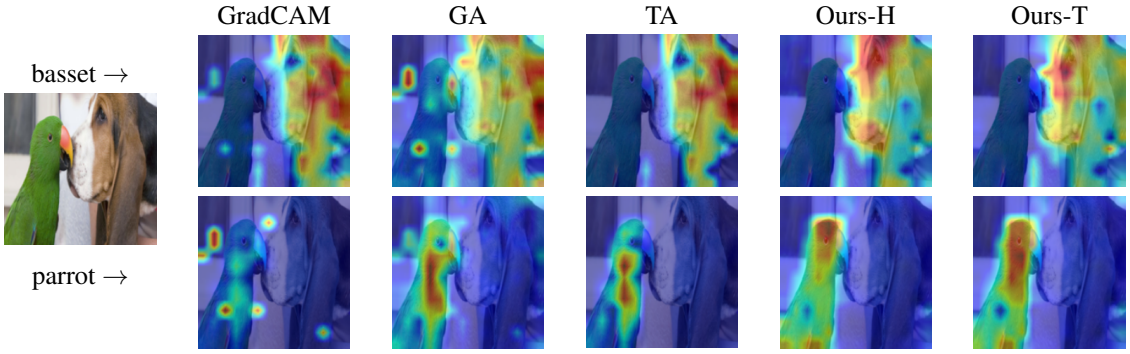

Figure 5: Class-specific visualization results of ViT-Base. We select the class discriminative methods at the very first place, and present the results of two different classes.

CLIP (Radford et al., 2021) for bi-modality. And we will present the experimental results by models. In terms of a Transformer literature and model size equivalence, we select BERT-base, ViT-base-16-224 and ViT-B/32 for CLIP, where all of them are derived from the Transformer Encoder architecture. The architectures of BERT and ViT are the same, except that BERT deals with (sub-)words while ViT copes with patches of an input image. CLIP adopts a dual encoder, in which text and image are respectively encoded and interact by cosine similarities.

ViT. Class-discriminability and the capacity of capturing fine-grained details are the criteria of a good visual explanation (Selvaraju et al., 2017). Figures 4 and 5 provide the visualizations of some images selected from ImageNet (Deng et al., 2009), demonstrating that our method is of such capacities. More samples can be found in appendix. As shown in the figures, Rollout captures a complete region of the target, but simply taking an average of attentions brings noises on the irrelevant pixels. TA and GA eliminate all the attentions with negative gradients which may result

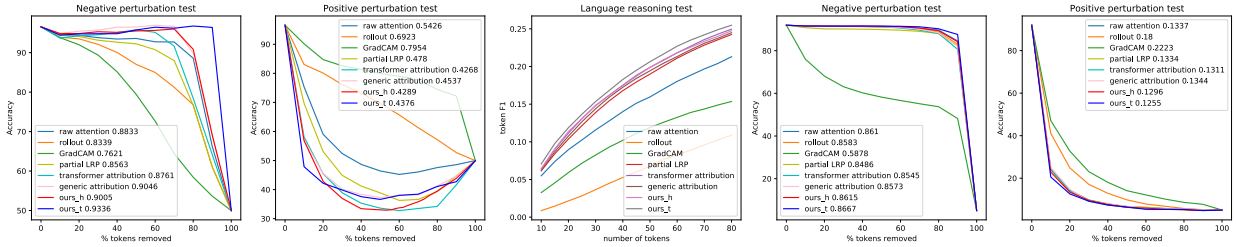

Figure 6: Results for BERT: *(from left to right)* Negative Perturbation Test on Movie Reviews, Positive Perturbation Test on Movie Review, Language Reasoning Test on Movie Reviews, Negative Perturbation Test on 20 Newsgroups and Positive Perturbation Test on 20 Newsgroups.

Table 1: Comparison of Positive (lower is better) and Negative (higher is better) perturbation AUC scores (percents) on ImageNet for ViT-Base, ViT-Large and ViT-MAE. H stands for the Head-Wise methods and T for Token-Wise.

			RAM	Rollout	CAM	PLRP	GA	TA	Ours-H	Ours-T
ViT-Base	Positive	Predicted	24.51	20.45	34.61	20.18	20.95	17.37	15.97	**15.89**
		Target	-	-	34.07	20.20	20.01	16.65	15.14	**15.13**
	Negative	Predicted	45.65	53.86	41.95	50.62	48.95	54.53	57.13	**57.97**
		Target	-	-	42.46	50.64	51.23	55.70	58.54	**59.24**
ViT-Large	Positive	Predicted	27.71	21.91	45.66	30.19	19.21	20.41	18.01	**17.99**
		Target	-	-	45.49	30.19	18.53	19.93	17.80	**17.57**
	Negative	Predicted	40.99	53.44	47.58	37.14	54.72	52.67	55.86	**56.44**
		Target	-	-	47.69	37.16	56.13	53.25	56.33	**56.88**
ViT-MAE	Positive	Predicted	38.56	38.20	56.79	26.59	33.11	34.00	**20.57**	20.72
		Target	-	-	57.24	26.57	32.63	33.74	**20.22**	20.30
	Negative	Predicted	40.79	52.03	24.80	55.34	57.67	56.92	**65.33**	64.48
		Target	-	-	24.41	55.37	58.80	57.31	**66.06**	65.18

Table 2: Segmentation test (percent, higher is better) on the ImageNet-Segmentation dataset for ViT-Base, ViT-Large and ViT-MAE.

		RAM	Rollout	CAM	PLRP	GA	TA	Ours-H	Ours-T
ViT-Base	mIoU	46.37	55.42	41.30	57.95	58.34	61.92	60.74	**66.32**
	pixel accuracy	67.87	73.54	65.91	76.31	76.30	79.68	78.04	**82.15**
	mAP	80.24	84.76	71.60	84.67	85.28	85.99	86.18	**88.04**
	mF1	29.44	43.68	19.42	38.82	41.85	40.10	44.45	**45.72**
ViT-Large	mIoU	41.18	52.88	39.72	40.09	54.40	56.31	61.24	**61.65**
	pixel accuracy	63.20	71.15	68.49	62.31	73.93	75.75	**78.92**	78.87
	mAP	74.75	83.48	63.29	73.56	81.93	83.39	85.52	**86.33**
	mF1	25.58	42.76	10.25	23.99	38.04	38.18	42.40	**43.38**
ViT-MAE	mIoU	37.30	42.39	22.60	51.13	47.66	45.98	**62.47**	62.36
	pixel accuracy	56.10	62.00	39.62	72.01	67.10	66.70	**79.66**	79.63
	mAP	67.41	76.38	57.28	81.15	79.58	78.41	**86.58**	86.21
	mF1	31.99	34.20	12.99	33.94	36.51	32.43	**44.67**	44.08

in an ignorance of some relevant pixels. Our methods perform well in capturing a very localized area without too much loss of the entire object, thanks to the combination of the attention perception that captures the entire object, and the reasoning feedback that decreases the noises of irrelevant pixels.

As for the quantitative evaluation, we select randomly 5 images per class (5k in total) from the ImageNet validation set for the perturbation tests, and the dataset of ImageNet-Segmentation (Guillaumin et al., 2014) for the segmentation tests. As shown in Tables 1 and 2, our methods outperform the other baselines on all metrics, especially the token-wise one, which leaves a great margin in segmentation tests, because of its more accurate approximation.

BERT. For the text modality, we first have the perturbation and language reasoning tests on the Movie Reviews Dataset (DeYoung et al., 2020) containing 1999 samples, which concerns a classification of positive and negative sentiments according to certain movie review. Our methods outperforms all the baselines in the language reasoning test.

For the perturbation test on Movie Reviews, the accuracy stops at a random guess (i.e. accuracy of 50%) at the end. We can see from Figure 6 that our methods outperform almost all the baselines, especially the token-wise method on the negative perturbation and language reasoning tests, where significant improvements are observed. As for the positive perturbation test, we see from the final score that TA is better than our methods, whereas in the first half of positive perturbation, the decrease according to both of our methods is clearly sharper than TA. We also notice that the methods

Table 3: Comparison of Positive (lower is better) and Negative (higher is better) perturbation AUC scores (percents) on ImageNet for CLIP.

		RAM	Rollout	GradCAM	GA	Ours-H	Ours-T
Positive	Predicted	14.11	19.32	21.49	12.78	**12.29**	12.32
	Target	-	-	21.34	11.91	11.55	**11.48**
Negative	Predicted	34.02	28.83	28.78	35.38	**35.74**	35.69
	Target	-	-	28.88	37.09	**37.21**	37.10

with good performance on image, such as Rollout and GradCAM, lose its advantage for text. In contrast, our methods stay competitive for both text and image.

Since both of quantitative test for BERT are based on the Movie Review Dataset and the same fine-tuned classifier, we have another text perturbation test on 20 Newsgroups Dataset (Lang, 1995) for category classification. The 20 Newsgroups Dataset is a a collection of approximately 20,000 newsgroup documents across 20 different newsgroups. We finetune a BERT-base model on its training data, with the accuracy reaching 93% on testing set. We randomly select 3000 documents from the testing set for the perturbation. The results are consistent with the perturbation test on Movie Review, as shown in Figure 6, where GradCAM and Rollout lose their advantages and our methods still outperform the other baselines. We provide as well the visualizations for BERT in the appendix, where our methods show more human-understandable explanation results without too much focus on tokens like “the” or “and”.

CLIP. A pretrained CLIP achieves good performances in some vision-language tasks, such as image captioning. Two visualization examples are shown in Figure 7. For images, the captured area of our method corresponds well with the text, even for multiple objects in a noun phrase. For texts, the captured tokens are also the key words in the noun phrase. More examples can be found in the appendix.

Besides, CLIP can also match the performance of original ResNet50 (He et al., 2016) on ImageNet for the zero-shot prediction task with a designed prompt, by calculating the similarities between “a photo of *class name*” and the corresponding image. For the perturbation test, we have a zero-shot image classification task with the same 5k images as ViT from the ImageNet validation set. As shown in Table 3, our methods outperform all the baselines as well.

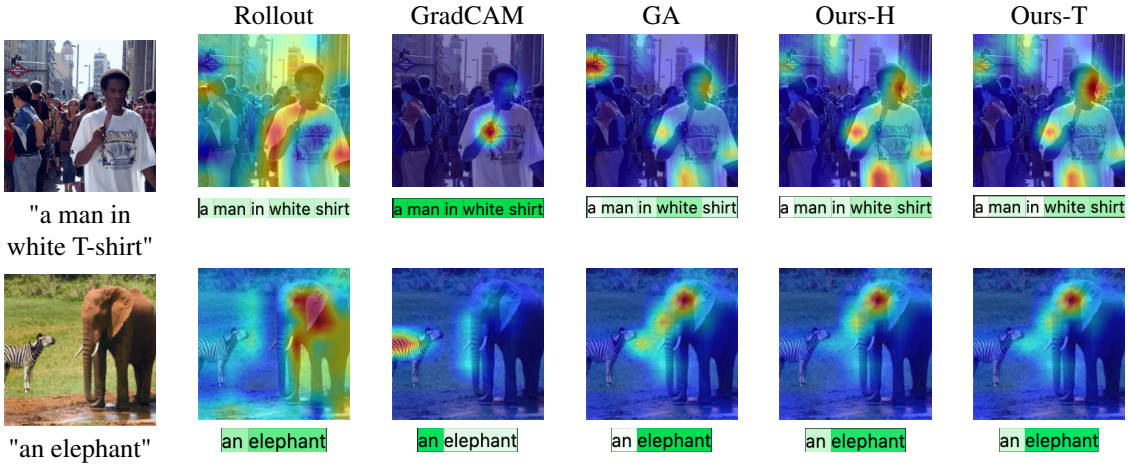

Figure 7: Visualization of bi-modal model: CLIP. The visualization results with our method correspond well with the highlighted input phrases, even for that with multiple objects.

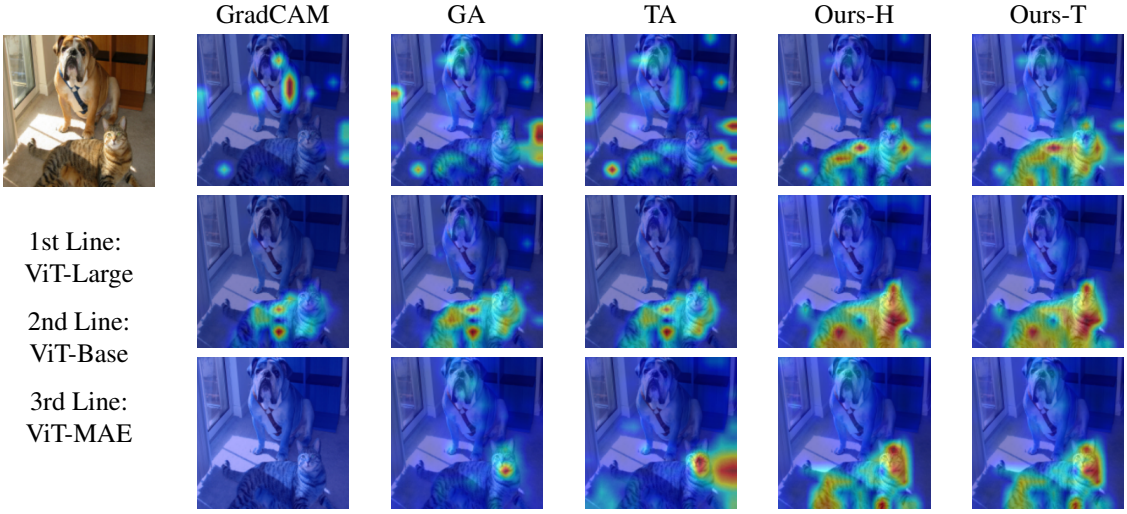

Figure 8: Visualizations of ViT-Large (1st line), ViT-Base (2nd line) and ViT-MAE (3rd line) on the same image for the same class *tiger cat*. Our methods show a more stable and less noisy capturing of semantic pixels than other baselines.

4.3 Transformers of Different Versions

In this subsection, we test the applicability of all explanation algorithms to different versions of Transformers, for example, with different model depth and pooling strategies. Besides **ViT-Base** in Sub-section 4.2, we conduct evaluation experiments on another two variants: ViT-large-16-224 (Dosovitskiy et al., 2021), denoted **ViT-Large** and **ViT-MAE** with the global pooling strategy instead of [CLS] pooling in vanilla ViT-base-16-224 model. ViT-Large (ViT-L/16) is a very big model, which consist of 24 successive attention blocks and 16 attention heads with embedding size set as 1024. ViT-MAE adopts the same architecture as ViT-Base, but with a different pretraining approach and pooling strategy. The masked autoencoder (MAE) (He et al., 2022) is a self-supervised learning approach of masking random patches of the input images and reconstructing the missing pixels. By reconstruction, the encoder learns more complex semantics, thus leading to better performances.

ViT-Large. We provide in the first line of Figure 8 the visualizations of ViT-Large for the class "tiger cat". It's interesting to see that the explanation methods, such as GradCAM, GA and TA, which capture the quite semantic tokens for ViT-base, focus on very irrelevant tokens for ViT-Large, whereas the ViT-Large shows better accuracy and transferability in the prediction. We believe that the irrationality results from an intuitive formulation of expression. With more successive attention blocks, the error accumulates, thus leading to less trustworthy visualizations. However, our methods, derived from explicit derivations, show far more understandable results in predicting the "tiger cat".

As for quantitative tests, we show in Table 2 and Table 1 the segmentation results and perturbation results for ViT-Large, where our methods outperform all the baselines in all the settings.

ViT-MAE. For all the baselines, their explanations of Transformer in global pooling setting are not provided in the original paper. To be more fair, we modify these methods by using the average of attribution scores from the second to the last line, indicating the first to the last token, instead of the first line for [CLS] token.

The visualization of ViT-MAE is shown in the third line of Figure 8 with the same photo and predicting class. Our methods not only capture the very semantic tokens, but also show a good coherence with the masked autoencoder approach, where by masking and reconstructing, the model can learn from more semantics. We credit the applicability of our methods in global pooling setting to the theoretical analysis as well.

Table 2 and Table 1 shows the results for segmentation and perturbation tests, where our methods leave a even more significant margin than the previous settings in both the positive and negative tests. It further demonstrates that our methods can also be applied to global pooled Transformers and provide trustworthy performances. Moreover, the strategy of masking and reconstructing in MAE can alleviate the out-of-distribution issue in the perturbation test (Hase et al., 2021; Hooker et al., 2019) and make the quantitative results more convincing.

4.4 Ablation Study

We propose two ablation studies: the first focuses on the effectiveness of our expression, while the second focuses on the benefits of introducing the token-wise and head-wise coefficients. We conduct our ablation analyses with the same evaluation settings for ViT-Base, mentioned in Section 4.2. From Table 4, we notice that the results correspond well with our derivations in Section 3. Introducing token-wise and head-wise attention maps clearly enhances the quality of explanations. Moreover, the first ablation study also indicates the effectiveness of introducing the integrated version of gradients to reduce the noise.

Table 4: Results of ablation study with ViT-Base. Avg stands for a simple average of attentions for all heads and $\mathbf{A}^{(\mathbf{L})}$ for the attention map of the last block. $\mathbf{G}^{(\mathbf{L})}$ denotes the attention gradient of the last block and $\int \mathbf{G}^{(\mathbf{L})}$ the integrated version. We report the results of perturbation test with its predicted class. The first part is designed to analyze the effectiveness our derived attribution expression $\mathcal{T}$ and the second part for the befits of introducing our token-wise/head-wise attention maps.

Setting		Segmentation				Perturbation	
Perception	Feedback	Acc.	mIoU	mAP	mF1	Pos.	Neg.
$\times$	$\int \mathbf{G}^{(\mathbf{L})}$	67.31	48.76	77.93	38.44	18.56	52.12
Head-wise	$\times$	71.83	53.24	83.66	42.26	20.20	52.79
Head-wise	$\mathbf{G}^{(\mathbf{L})}$	75.67	57.75	84.49	43.01	17.33	55.27
Head-wise	$\int \mathbf{G}^{(\mathbf{L})}$	**78.04**	**60.74**	**86.18**	**44.45**	**15.97**	**57.13**
$\mathbf{A}^{(L)}$	$\int \mathbf{G}^{(\mathbf{L})}$	77.55	59.16	84.65	38.82	19.05	52.45
Avg	$\int \mathbf{G}^{(\mathbf{L})}$	77.88	60.56	86.14	44.34	16.72	55.59
Head-wise	$\int \mathbf{G}^{(\mathbf{L})}$	78.04	60.74	86.18	44.45	15.97	57.13
Token-wise	$\int \mathbf{G}^{(\mathbf{L})}$	**82.15**	**66.32**	**88.04**	**45.72**	**15.89**	**57.97**

4.5 Token-wise/Head-wise

The token-wise and head-wise manners come from the different approximation approaches we use for the recurrence relation of each attention block, as discussed in Section 3. The token-wise one credits the effects of linear projection to the differences in the norms of token vectors while the head-wise one considers the head differences. As for implementation, the token-wise method may involve extra modifications to obtain the intermediates, since they are not the direct output of any component in the neural network. The head-wise one has no such issue because it only demands the attention and attention gradients for calculation.

However, both visualizations and quantitative results demonstrate that the differences between these two methods are very limited , since they are both derived within the same framework. They both outperform all the baseline methods and are applicable to different types of Transformers, which contributes to the explicitness behind these two methods.

5 Conclusions

Most of the Transformer-based models give the prediction by first integrating different tokens via successive attention blocks, and then using the pooled vector of last token representations for a prediction task. In this work, we propose a novel explanation method based on gradients, the change of basis and the chain rule, dividing the entire process into attention perception and reasoning processes. Moreover, by taking into account the linear projections, we derive two attention maps (token-wise and head-wise) for the attention perception process, which are more accurate than traditional Attention Rollout. We show in the experiments that our method achieves a better performance than all baselines in a series of evaluation settings, on specifying the class information, capturing the fine-grained pixels and being applicable to different model sizes, pooling strategies and modalities.

Based on this tool, we can explore in depth those Transformer-based models in the fields of both computer vision and language processing, which gives a hand in debugging, training and evaluating different Transformers. In this work, we only deal with Transformer Encoder. Other model architectures, such as Encoder-Decoder (Yang et al., 2019) or Decoder types (Brown et al., 2020), will be tackled in our future work.

Broader Impact Statement

Being explainable is an important criterion to implement the deep learning algorithms under real life settings. With the wide applications of Transformer-based models, our methods can be very helpful by bringing up a trustworthy explanation and increasing the transparency of model prediction. However, it may also be used for a malicious purpose if the user intends to trick the model with modifications to the most or the least important tokens.

References

Samira Abnar and Willem H. Zuidema. Quantifying attention flow in transformers. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 4190–4197. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.385.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *CoRR*, abs/1607.06450, 2016.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B. Tenenbaum, William T. Freeman, and Antonio Torralba. GAN dissection: Visualizing and understanding generative adversarial networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. In Alessandro E. P. Villa, Paolo Masulli, and Antonio Javier Pons Rivero (eds.), *Artificial Neural Networks and Machine Learning - ICANN 2016 - 25th International Conference on Artificial Neural Networks, Barcelona, Spain, September 6-9, 2016, Proceedings, Part II*, volume 9887 of *Lecture Notes in Computer Science*, pp. 63–71. Springer, 2016. doi: 10.1007/978-3-319-44781-0_8.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 9630–9640. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00951.

Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pp. 782–791. Computer Vision Foundation / IEEE, 2021a.

Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 387–396. IEEE, 2021b. doi: 10.1109/ICCV48922.2021.00045.

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 9620–9629. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00950.

Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter F. Stewart. RETAIN: an interpretable predictive model for healthcare using reverse time attention mechanism. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 3504–3512, 2016.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pp. 248–255. IEEE Computer Society, 2009. doi: 10.1109/CVPR.2009.5206848.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 4443–4458. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.408.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Matthieu Guillaumin, Daniel Küttel, and Vittorio Ferrari. Imagenet auto-annotation with segmentation propagation. *Int. J. Comput. Vis.*, 110(3):328–348, 2014. doi: 10.1007/s11263-014-0713-9.

Yaru Hao, Li Dong, Furu Wei, and Ke Xu. Self-attention attribution: Interpreting information interactions inside transformer. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 12963–12971. AAAI Press, 2021.

Peter Hase, Harry Xie, and Mohit Bansal. The out-of-distribution problem in explainability and search methods for feature importance explanations. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 3650–3666, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 770–778. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.90.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 15979–15988. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01553.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 9734–9745, 2019.

Drew A Hudson and Christopher D Manning. Gqa: a new dataset for compositional question answering over real-world images. *arXiv preprint arXiv:1902.09506*, 3(8), 2019.

Sarthak Jain and Byron C Wallace. Attention is not explanation. 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. 3543–3556, 2019.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attention is not only a weight: Analyzing transformers with vector norms. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pp. 7057–7075. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.574.

Ken Lang. Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pp. 331–339, 1995.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *CoRR*, abs/1908.03557, 2019.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 9992–10002. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00986.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 14014–14024, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (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. 8748–8763. PMLR, 2021.

Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi (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.

Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. 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.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pp. 618–626. IEEE Computer Society, 2017. doi: 10.1109/ICCV.2017.74.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda B. Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *CoRR*, abs/1706.03825, 2017.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In Doina Precup and Yee Whye Teh (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.

Hao Tan and Mohit Bansal. LXMERT: learning cross-modality encoder representations from transformers. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pp. 5099–5110. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1514.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 5998–6008, 2017.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In Anna Korhonen, David R. Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 5797–5808. Association for Computational Linguistics, 2019. doi: 10.18653/v1/p19-1580.

Minh N. Vu, Truc D. T. Nguyen, NhatHai Phan, Ralucca Gera, and My T. Thai. Evaluating explainers via perturbation. *CoRR*, abs/1906.02032, 2019.

Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pp. 111–119. Computer Vision Foundation / IEEE, 2020. doi: 10.1109/CVPRW50498.2020.00020.

Sarah Wiegreffe and Yuval Pinter. 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.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. VLM: task-agnostic video-language model pre-training for video understanding. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pp. 4227–4239. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.findings-acl.370.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In Francis R. Bach and David M. Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 2048–2057. JMLR.org, 2015.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 5754–5764, 2019.

Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnnexplainer: Generating explanations for graph neural networks. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 9240–9251, 2019.

Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 2921–2929. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.319.

A Table of Notations

Table 5: Table of notations

Indices:	
L	number of multi-head self-attention blocks ($l \in \{1, ..., L\}$), which is used as the superscript (l) to denote the variables and parameters concerning the l-th block.
N	number of tokens/patches ($i \in \{0, 1, ..., N\}$ and 0 denotes the added [CLS] token), where the subscript (i, j) stands for the $(i+1)$-th line and $(j+1)$-th column of the matrix, representing the i-th and j-th tokens respectively and $(i, \cdot)$ for the entire $(i+1)$-th line.
C	number of classes during the prediction ($c \in \{1, ..., C\}$)
H	number of heads in the self-attention block ($h \in \{1, ..., H\}$)
K	number of steps while discretizing the integration ($k \in \{1, ..., K\}$), which is only involved in Section 3.3.
$\mathcal{P}$	the pooling strategy ($\mathcal{P} \in \{\mathrm{cls}, \mathrm{glp}, \dots\}$)
Our Explanation Method:	
$\mathcal{T}$	expression of our explanation method
$\mathbf{P}$	attention perception
$\mathbf{F}$	reasoning feedback
$\tilde{\mathbf{A}}_{\mathrm{token}}$	the token-wise attention map
α	the token-wise coefficients
$\tilde{\mathbf{A}}_{\mathrm{head}}$	the head-wise attention map
I	the head importance
θ	the head-wise coefficients
Transformer Parameters:	
$\mathbf{E}$	the embedding module
$\mathbf{E}_{\mathrm{pos}}$	the positional embedding
D	the embedding size
D_{hidden}	the hidden size
W	the linear transformation matrix
b	the bias in linear transformation
Variables:	
X	the input image patches or text tokens
$\mathbf{Z}$	the input/output of self-attention block
$\tilde{\mathbf{Z}}$	the output after multi-head self-attention module
$\mathbb{B}$	the set of basis vectors representing each token
$\tilde{X}$	the basis vector
$\tilde{x}$	the coordinate of input vector over basis vector
$\mathbf{A}$	the raw attention map
Q	query
K	key
V	value
$\mathcal{L}$	the loss function

B Details of Baselines

Raw attention map (RAM) Raw attention map is a rather straightforward explanation method, which takes into consideration only the attention map of the last block. For the model with a [CLS] pooling method, it explains the importance of different patches with the first line of the attention matrix. In our experiment, the last attention layer is visualized.

Attention Rollout (Abnar & Zuidema, 2020) (Rollout) Different from raw attention, all the attention maps along a forward-pass count for the explanation, where they assume a linear combination of tokens and quantify the influence of skip connections by adding an identity matrix. Note that both raw attention and Attention Rollout are class-agnostic.

GradCAM (Selvaraju et al., 2017) GradCAM provides a class-specific explanation, where they add weights to gradient based feature map of the last convolution layer for a ConvNet based model. Following the work of Chefer et al. (2021a), we use a weighted gradient map of the last attention block, which corresponds to the [CLS] token .

Partial LRP (Voita et al., 2019) (PLRP) Not only the attention map of each block is considered, the relevance propagation based methods also take the information flow inside the whole neural network into account. These methods are also class-agnostic, which means the visualization remains the same for the prediction of all classes. Here, we visualize the relevance map of the last attention layer.

Transformer attribution (Chefer et al., 2021a) (TA) Transformer attribution method is a state-of-the-art class-specific explanation method for Transformer. It combines relevancy and attention-map gradient by regarding the gradient as a weight to the relevance for certain prediction task.

Generic attribution (Chefer et al., 2021b) (GA) Generic attribution extends the usage of Transformer attribution to co-attention and self-attention based models, such as VisualBERT, LXMERT etc. and propose a more generic relevancy update rule. Meanwhile, it replaces the relevance of Transformer attribution with the attention layer in each block.

C Token-wise Attention Map

Here, we provide the mathematical derivation details of the Eq. (14) and Eq. (15). The goal is to develop the formula $\mathbf{O} = \mathbf{A}\mathbf{Z}W$, where $\mathbf{O} \in \mathbb{R}^{(N+1)\times D}$, $\mathbf{A} \in \mathbb{R}^{(N+1)\times(N+1)}$ and $W \in \mathbb{R}^{D\times D}$.

By the definition of matrix product, we have:

$$\mathbf{O}_{i,j} = \sum_{m=1}^{N+1}\sum_{k=1}^{D} \mathbf{A}_{i,m}\mathbf{Z}_{m,k}W_{k,j} \quad . \tag{19}$$

Since $\mathbf{A}_{i,m}$, $\mathbf{Z}_{m,k}$ and $W_{k,j} \in \mathbb{R}$, it is possible to write when $\mathbf{Z}_{m,j} \neq 0$ that,

$$\begin{aligned}
\mathbf{O}_{i,j} &= \sum_{m=1}^{N+1}\sum_{k=1}^{D} \mathbf{A}_{i,m}\frac{\mathbf{Z}_{m,k}W_{k,j}}{\mathbf{Z}_{m,j}}\mathbf{Z}_{m,j} = \sum_{m=1}^{N+1}(\sum_{k=1}^{D} \mathbf{A}_{i,m}\frac{\mathbf{Z}_{m,k}W_{k,j}}{\mathbf{Z}_{m,j}})\mathbf{Z}_{m,j} && (20)\\
&= \sum_{m=1}^{N+1}\frac{\mathbf{A}_{i,m}}{\mathbf{Z}_{m,j}}(\sum_{k=1}^{D}\mathbf{Z}_{m,k}W_{k,j})\mathbf{Z}_{m,j} = \sum_{m=1}^{N+1}\frac{\mathbf{A}_{i,m}}{\mathbf{Z}_{m,j}}(\mathbf{Z}W)_{m,j}\mathbf{Z}_{m,j} && (21)\\
&= \sum_{m=1}^{N+1}\mathbf{A}_{i,m}\frac{(\mathbf{Z}W)_{m,j}}{\mathbf{Z}_{m,j}}\mathbf{Z}_{m,j} \underset{\text{wise}}{\overset{\text{token}}{\approx}} \sum_{m=1}^{N+1}\mathbf{A}_{i,m}\frac{\|(\mathbf{Z}W)_{m,\cdot}\|}{\|\mathbf{Z}_{m,}\|}\mathbf{Z}_{m,j} && (22)\\
&= \sum_{m=1}^{N+1}\tilde{\mathbf{A}}_{i,m}\mathbf{Z}_{m,j} = (\tilde{\mathbf{A}}_{\text{token}}\mathbf{Z})_{i,j} \quad . && (23)
\end{aligned}$$

Here comes the definition of our token-wise attention map.

D Discussion about Self-Supervised Learning

Self-supervised learning strategy can bring new properties to the model, especially to the Transformer-based models (Caron et al., 2021; Chen et al., 2021; He et al., 2022). DINO (Caron et al., 2021) is a self-supervised learning method for Vision Transformer, with a specific design to reflect directly semantics via features. In such case, adopting simple explanation methods such as raw attention map might already be promising. In order to understand the differences

that the self-supervised learning strategies can bring to explanation results, we add an additional segmentation test for ViT-Base-DINO on ImageNet-Segmentation (Guillaumin et al., 2014). We apply our token-wise/head-wise methods and compare them with raw attention map.

Table 6: Segmentation test on ImageNet-Segmentation dataset (percent) for ViT-DINO

		Raw Attention Map	Ours-H	Ours-T
ViT-Base-DINO	IoU	65.79	**67.42**	66.42
	Acc	82.08	**83.71**	83.09
	mAP	86.79	**87.60**	87.26

As shown in Table 6, self-supervised learning can bring differences in explaining certain models, whereas our methods still perform better on localizing semantics. We notice that the results of our methods for DINO is almost the same as Supervised ViT-Base in Table 2. This may indicate that there are some hidden and hard-to-get semantics in the supervised ViT-Base that are equal to ViT-DINO. And our methods succeed in revealing this information and being applicable to all Transformer-based models.

Moreover, the ImageNet-Segmentation dataset only contains images with single objects. For explanation methods that are not class-specific, the segmentation results can be ambiguous when it appears multiple objects on the image. We show in Figure 9 the visualizations for multiple objects, which further validate the benefits of our methods in capturing semantics.

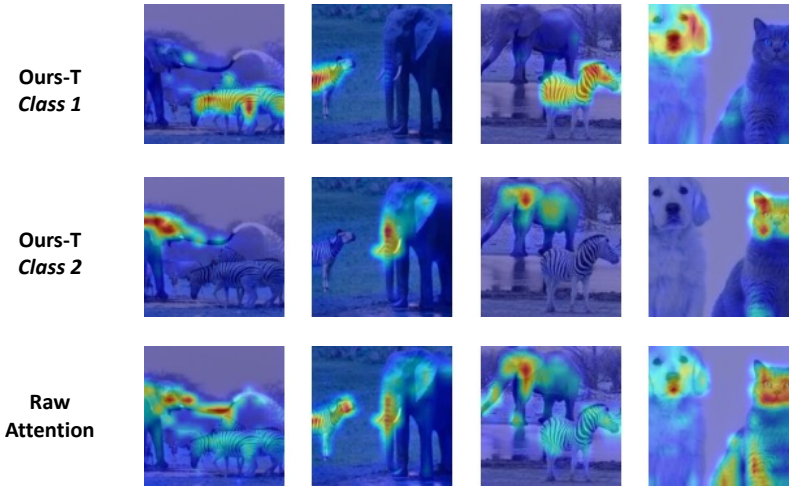

Figure 9: Class-specific visualizations for ViT-Base-DINO.

E More Samples, BERT and ViTs (Supervised, MoCo-v3, DINO and MAE)

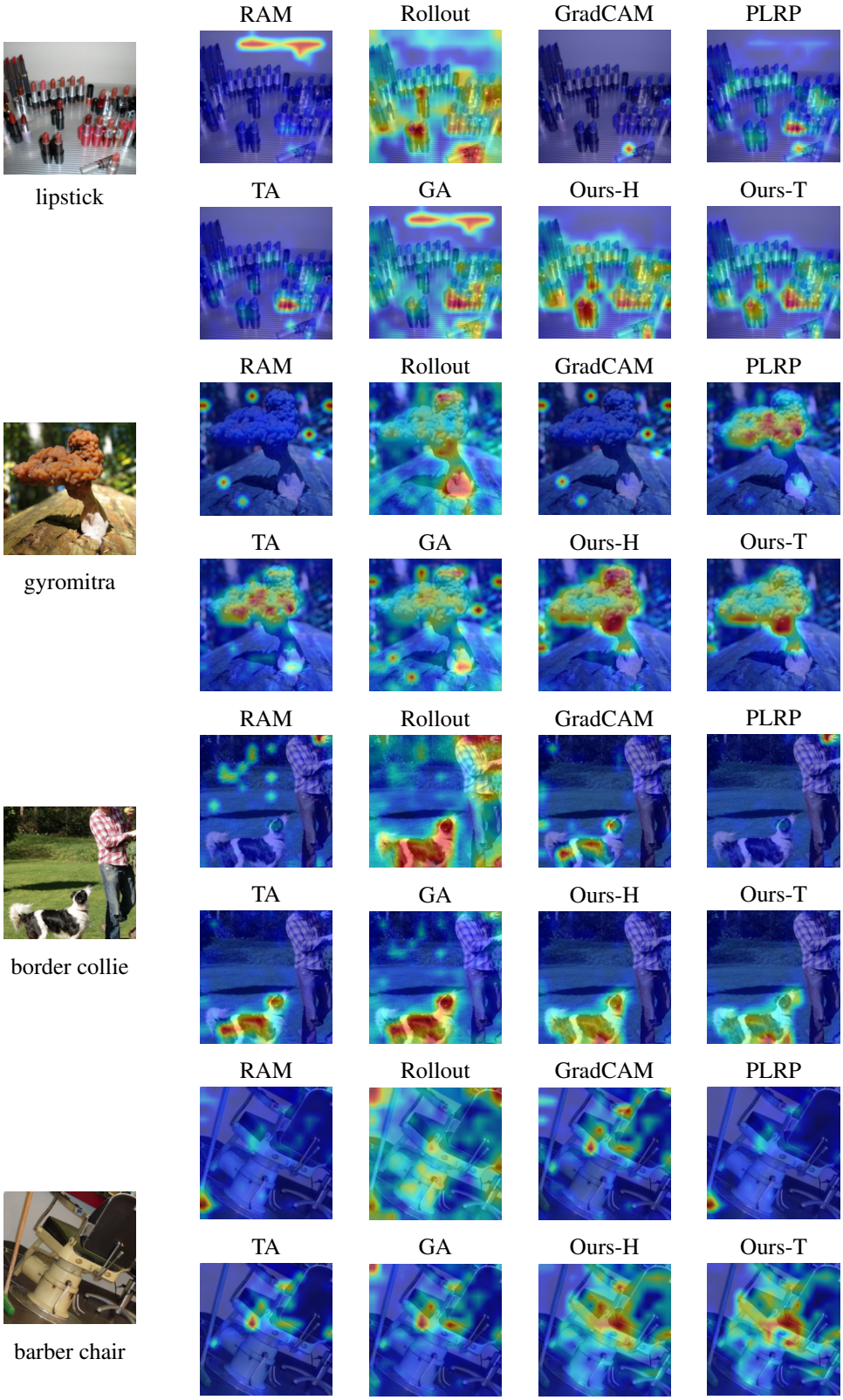

Figure 10: More samples of localization of fine-grained regions for single class prediction of ViT with full baselines.

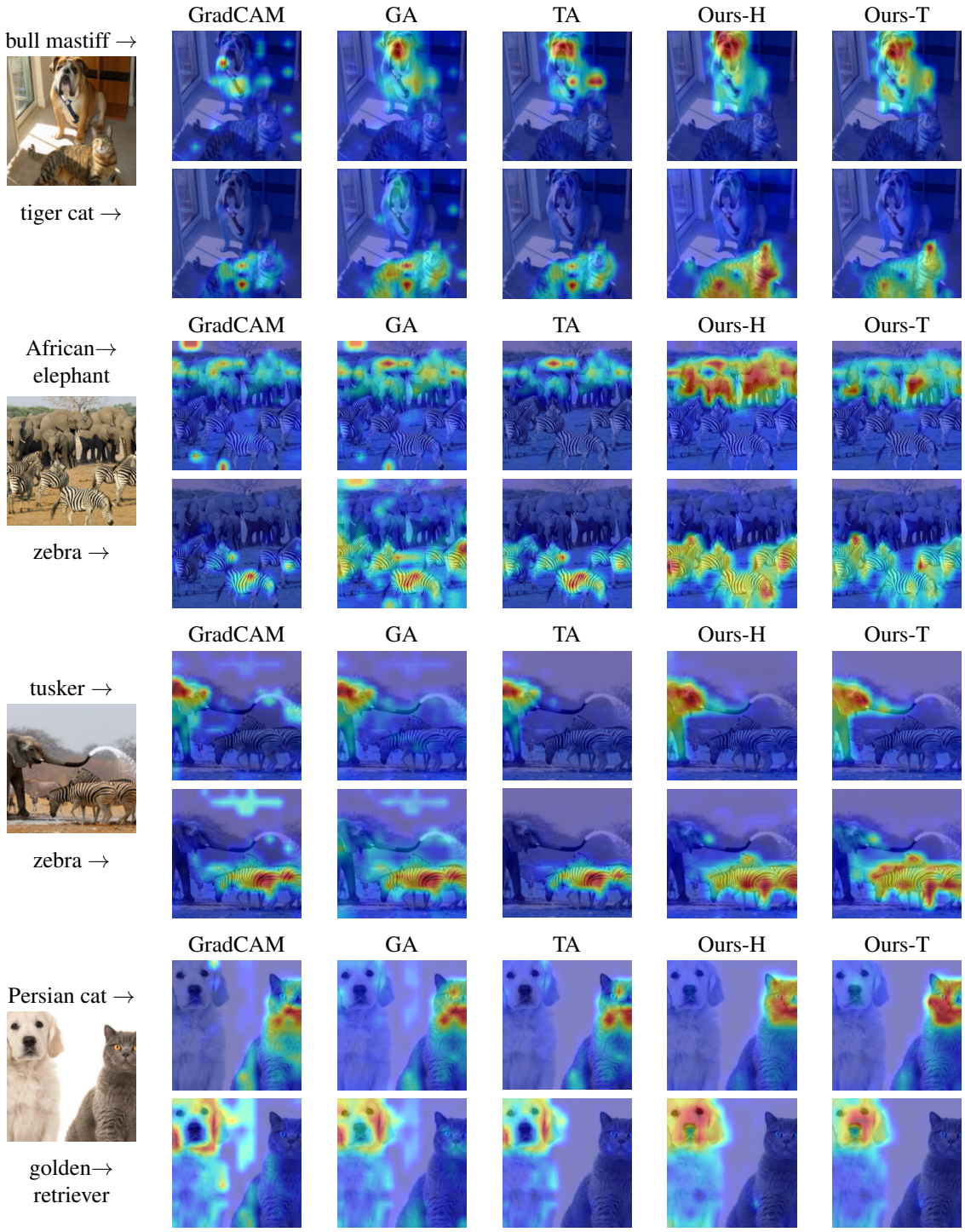

Figure 11: More samples of class-specific visualization results of ViT. We just present the results of two classes with class-specific baseline methods, while for the other methods, there are not any differences between different classes.

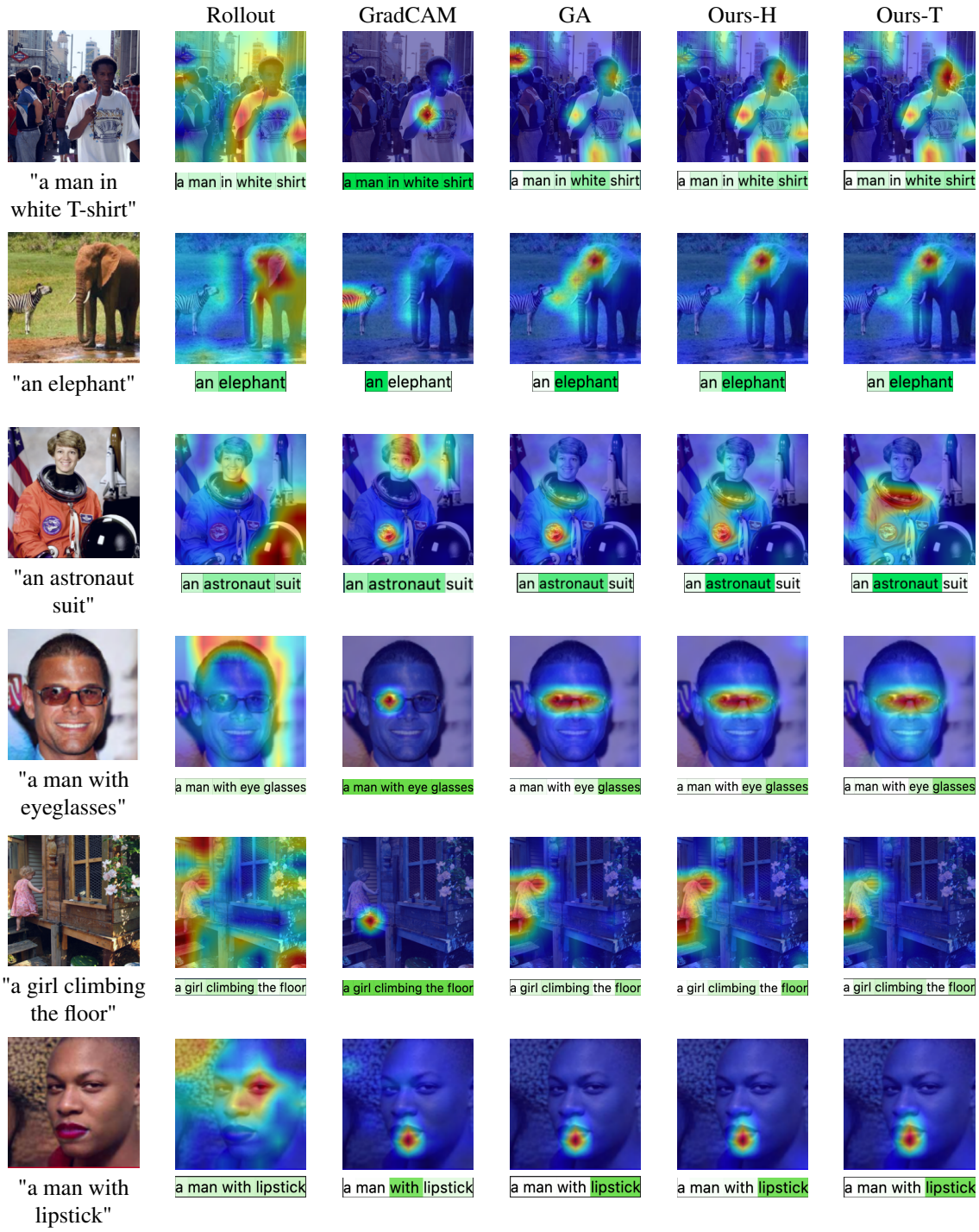

Figure 12: More samples of CLIP visualization. Both text and image visualizations are provided here.

Figure 13: BERT visualizations for a sample with negative emotion in Movie Reviews Dataset. A darker color represents a greater token attribution during the prediction.

Figure 14: BERT visualizations for a sample with positive emotion in Movie Reviews Dataset. A darker color represents a greater token attribution during the prediction.

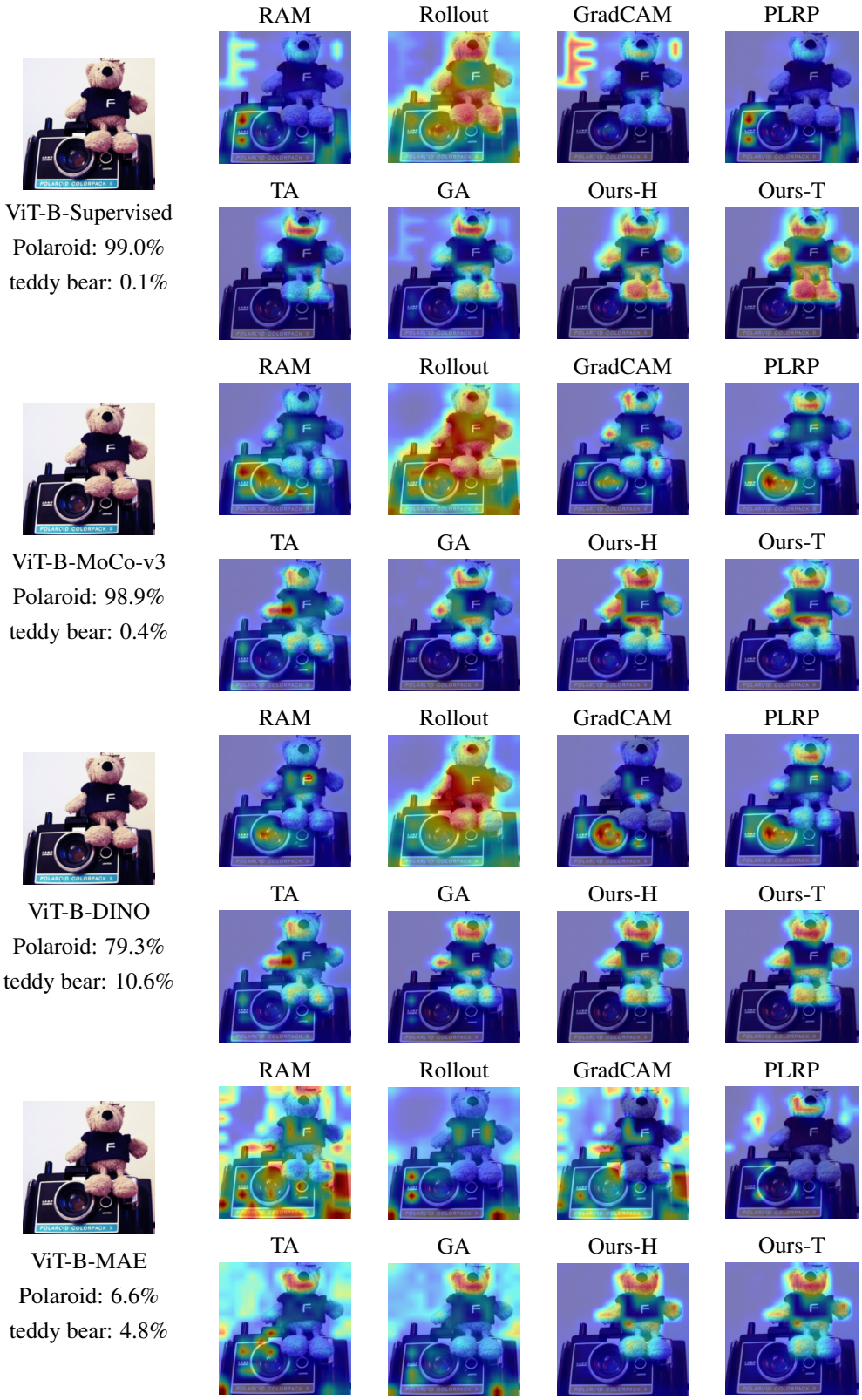

Figure 15: Visualizations *class: teddy, teddy bear* for ViT-Base with different self-supervised learning (SSL) strategies: MoCo-v3 (Chen et al., 2021), DINO (Caron et al., 2021) and MAE (He et al., 2022). Although SSL can influence explanations, our methods show stable performances and good consistency with models' properties.