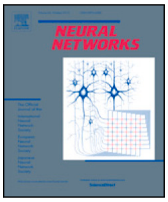
Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet



Full length article

GradToken: Decoupling tokens with class-aware gradient for visual explanation of Transformer network

Lin Cheng [a], Yanjie Liang [b], Yang Lu [a,*], Yiu-ming Cheung [c]

[a] *Fujian Key Laboratory of Sensing and Computing for Smart City, School of Informatics, Xiamen University, Xiamen 361005, China*
[b] *Peng Cheng Laboratory, Shenzhen 518000, China*
[c] *Department of Computer Science, Hong Kong Baptist University, Hong Kong Special Administrative Region of China*

ARTICLE INFO

Keywords:
Explanation
Interpretability
Visualization
Transformer

ABSTRACT

Transformer networks have been widely used in the fields of computer vision, natural language processing, graph-structured data analysis, etc. Subsequently, explanations of Transformer play a key role in helping humans understand and analyze its decision-making and working mechanism, thereby improving the trustworthiness in its real-world applications. However, it is difficult to apply the existing explanation methods for convolutional neural networks to Transformer networks, due to the significant differences between their structures. How to design a specific and effective explanation method for Transformer poses a challenge in the explanation area. To address this challenge, we first analyze the semantic coupling problem of attention weight matrices in Transformer, which puts obstacles in providing distinctive explanations for different categories of targets. Then, we propose a gradient-decoupling-based token relevance method (i.e., GradToken) for the visual explanation of Transformer's predictions. GradToken exploits the class-aware gradient to decouple the tangled semantics in the class token to the semantics corresponding to each category. GradToken further leverages the relations between the class token and spatial tokens to generate relevance maps. As a result, the visual explanation results generated by GradToken can effectively focus on the regions of selected targets. Extensive quantitative and qualitative experiments are conducted to verify the validity and reliability of the proposed method.

1. Introduction

As a recent research hotspot in the deep neural network family, Transformer (Chu et al., 2021; Han et al., 2023; Liu et al., 2021) has gained significant attention, particularly when applications based on GPT (Generative Pre-trained Transformer) (Brown et al., 2020) models enter our daily lives. Transformer was firstly proposed in the field of natural language processing (Vaswani et al., 2017). With the proposed Vision Transformer (ViT) (Dosovitskiy et al., 2021) in 2021, Transformers have been widely used in object recognition (Vasanthi & Mohan, 2023), anomaly detection (Chen, You, Zhang, Xi, & Le, 2022), and image segmentation (Zhang et al., 2024), just to name a few.

In the literature, a number of works, e.g., see Carion et al. (2020), Cheng, Liu, Fan, Feng, and Jia (2024), Yuan, Hou, Jiang, Feng, and Yan (2023), focus on application research on Transformers. However, Transformers suffer from the black-box dilemma, which limits their applications in the critical areas (Qiang, Pan, Li, Li, Jang, & Zhu, 2022), such as autonomous driving, medical diagnosis, and digital finance. Unfortunately, addressing explanatory research (Chefer, Gur, & Wolf, 2021b) on Transformers has not been well studied yet. Obviously, the explanatory research on Transformer provides valuable insights for understanding its structure, guiding its analysis and design (Ma et al., 2023), supporting its practical applications in critical scenarios (Qiang et al., 2022), and assisting its utilization in other tasks (Xu, Ouyang, Bennamoun, Boussaïd, & Xu, 2022).

The structure of Transformer networks differs significantly from that of convolutional networks (CNNs) (He, Zhang, Ren, & Sun, 2016; Simonyan & Zisserman, 2015). CNN mainly consisting of convolutional layers and pooling layers, performs each feature extraction within a local region. Different from CNN, Transformer consists of alternating self-attention modules and multi-layer perceptron (MLP) modules as its core components, which can encode or decode token variables. Compared to CNN, information inside Transformer can be interacted globally. Additionally, Transformer contains the structures such as patch embedding, positional encoding, class token, and GELU activation layers (Hendrycks & Gimpel, 2016) (which allow the activated tokens containing negative values), which increase the specificity and

[*] Corresponding author.
E-mail addresses: cheng.charm.lin@hotmail.com (L. Cheng), liangyj@pcl.ac.cn (Y. Liang), luyang@xmu.edu.cn (Y. Lu), ymc@comp.hkbu.edu.hk (Y.-m. Cheung).

https://doi.org/10.1016/j.neunet.2024.106837
Received 3 March 2024; Received in revised form 18 June 2024; Accepted 19 October 2024
Available online 1 November 2024

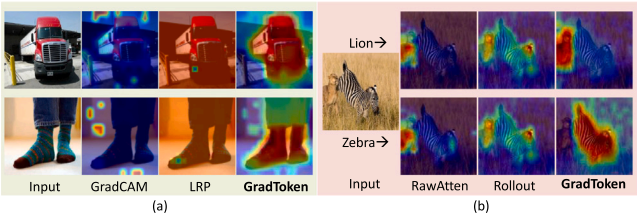


Fig. 1. Visualizations for the model predictions via running the baselines GradCAM (Selvaraju, Cogswell, Das, Vedantam, Parikh, & Batra, 2020), LRP (Bach et al., 2015), RawAtten (Clark, Khandelwal, Levy, & Manning, 2019), Rollout (Abnar & Zuidema, 2020), and the proposed GradToken. (a) GradCAM and LRP fail to highlight the target regions. (b) Both RawAtten and Rollout cannot discriminate different targets. In contrast, GradToken can focus on the targets and have better selectivity.

complexity of its structure. Furthermore, the output layer of Transformer (e.g. ViT (Dosovitskiy et al., 2021)) is not processed by global pooling (as in CNN), instead, the class token is chosen and fed into the final linear classification layer, resulting in the other tokens in the last layer being untrained. The specificity and complexity of the Transformer network increase the difficulty of its explanatory study.

In recent years, some visual explanation methods, such as GradCAM (Selvaraju et al., 2020) and LRP (Bach et al., 2015), have been proposed to explain CNNs. However, when directly applied to explain the Transformer network, GradCAM and LRP may produce visualizations that highlight background regions unrelated to the model's decisions (as shown in Fig. 1(a)). In contrast to directly applying explanation methods designed for CNNs, some works (e.g., RawAtten (Clark et al., 2019; Kovaleva, Romanov, Rogers, & Rumshisky, 2019)) utilize the similarity matrix obtained from attention computation in Transformer as the explanation. This similarity matrix tends to highlight foreground regions where potential targets may present. Another enhanced method called Rollout (Abnar & Zuidema, 2020) combines multiple layers of attention weights and propagates the attention from higher layers to lower layers using matrix multiplication. However, both RawAtten and Rollout suffer from reliability issues in their explanations. Specifically, these methods simply focus on foreground regions without the ability to differentiate between different targets (as shown in Fig. 1(b)), resulting in a lack of selectivity.

To address the limited reliability of current visual explanation methods for Transformer networks, we revisit the computational process of attention and analyze the information interaction between the class token and the other tokens. In Transformer, the class token is not directly supervised by category-specific information, resulting in a lack of association with different semantic categories. As a result, the similarity computed between the class token and the other tokens can only distinguish foreground and background regions, but cannot discriminate regions of different categories. Therefore, a pivotal point of our investigation lies in decoupling the class token to align with the semantics of different categories without changing the network structure when explaining the vision Transformer.

In this paper, we propose a gradient-decoupling-based token relevance method named GradToken for visualizing and explaining the vision Transformer. In the proposed GradToken, we first select a target class in the output layer and compute the gradient of the prediction score with respect to the class token. Then, the similarity between the gradient of the class token and other spatial tokens is computed, generating a corresponding relevance map. By decoupling the gradient of the class token from the tangled semantics of multiple classes, the generated relevance map can focus on the selected target class. Furthermore, we investigate how to exploit information across multilevel attention layers to propagate the relevance map from higher layers to lower layers. Specifically, we utilize the similarity matrix from lower layers to refine the relevance maps from higher layers, leading to superior explanatory results. To verify the effectiveness of the proposed method, segmentation experiments are conducted on the ImageNet-Segmentation dataset (Guillaumin, Küttel, & Ferrari, 2014) and the PASCAL VOC dataset (Everingham, Gool, Williams, Winn, & Zisserman, 2010). Perturbation experiments are performed on the ImageNet classification dataset (Russakovsky et al., 2015). The experimental results demonstrate that the proposed GradToken achieves higher accuracy and better reliability compared to the current explanation methods.

The main contributions of this paper are summarized as follows:

- A gradient-based semantic decoupling method is proposed to decouple the tangled semantics of class token. It can associate the class token with the semantic information of the target category to improve the target selectivity of the explanation. This semantic decoupling process does not require changing the original structure of Transformer.
- A category relevance computation method is proposed to compute the relevance of spatial tokens to different categories. It can effectively generate the relevance map of each category for the visual explanation of Transformer.
- The effects of different attention aggregation and propagation schemes, multi-head relevance integration schemes, and relevance propagation depths on the explanation results are fully studied. Extensive experiments show the superiority of the proposed method.

2. Related work

An increasing number of methods have been introduced to explain Transformer networks, especially to reveal the important features for the decision-making of the networks in a visualization way. In the following, we introduce explanation methods based on attention, relevance, gradients, integrated gradients, etc.

2.1. Attention-based visual explanation methods

Considering that the self-attention mechanism in Transformers builds relevances between pairwise tokens by assigning scores, RawAtten was proposed to explain Transformer models based on the attention mechanism (Clark et al., 2019; Kovaleva et al., 2019). Its core idea is to use attention weights to evaluate the relevance between the model's output tokens and input tokens. However, Serrano and Smith (2019) found that erasing the strongly responsive regions in the visualizations generated by RawAtten does not have a significant impact on the model's performance. The findings indicate that the explanation results obtained by RawAtten are not reliable.

As an improvement of RawAtten, Abnar and Zuidema (2020) proposed Rollout to probe the flow of crucial information from the input layer to the higher layers of the network, and computes the relevances among all layers and positions through consecutive matrix multiplications. However, both RawAtten and Rollout suffer from a common issue, namely, the lack of associating the visual explanations with specific categories, leading to low reliability of the explanation results.

2.2. Relevance-based visual explanation methods

Layer-wise Relevance Propagation (LRP) (Bach et al., 2015) was also applied to explain the decisions of Transformer models (Chefer et al., 2021b). LRP is used to propagate the relevance of the output token layer by layer back to the input tokens. Building upon LRP, Voita, Talbot, Moiseev, Sennrich, and Titov (2019) proposed a method called Partial Layer-wise Relevance Propagation (PLRP), which captures the relative importance of attention heads in each Transformer encoder block. However, PLRP only considers the local information of relevance for each attention head and does not propagate the relevance scores to the input tokens.

Chefer et al. (2021b) proposed TransAttrib to propagate the target relevance from the output token layer by layer to the input tokens based on the Deep Taylor Decomposition (Montavon, Lapuschkin, Binder, Samek, & Müller, 2017) rule. TransAttrib integrates both the gradients of attention weights and relevance scores, and combines the integrated results from multiple attention modules. However, TransAttrib exhibits errors in visualizations generated for lower layers of the network. The visualizations generated by this method may neglect some discriminative features.

More recently, Vukadin, Afrić, Šilić, and Delač (2024) proposed absLRP to both explain CNN and Transformer. To generate sparse and contrastive visualization maps, absLRP discards the negative parts from neurons' contributions and adopts the absolute final output of neurons for normalization. However, the generated maps may be too sparse, which neglect some important features.

2.3. Gradient-based visual explanation methods

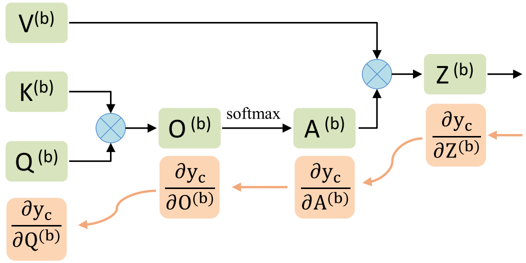
Based on TransAttrib (Chefer et al., 2021b), Chefer, Gur, and Wolf (2021a) further proposed a generic explanation method to explain a multimodal encoder–decoder Transformer model. Instead of using layer-wise relevance propagation, this method adopts a calculation similar to GradCAM (Selvaraju et al., 2020) to obtain the visual explanatory results. Compared to TransAttrib, this method offers a more concise computation. Subsequently, Qiang et al. (2022) proposed Attentive Class Activation tokens (AttCAT) to explain Transformers. AttCAT fully utilizes features, gradients, and attention weights to generate explanatory results. Recently, Leem and Seo (2024) introduced Attention Guided CAM (AGCAM) to advance the visual explanation of Transformer. AGCAM replaces $\mathrm{softmax}$ with $\mathrm{sigmoid}$ to normalize the attention weights to obtain better feature maps when computing the gradients-weighted feature maps.

There are also several methods that employ the integrated gradients (Sundararajan, Taly, & Yan, 2017) to explain Transformers. For example, Hao, Dong, Wei, and Xu (2021) proposed to calculate the integrated gradients for individual attention heads, and then compute the element-wise multiplication between the integrated gradients and the attention weight matrix. Yuan, Li, Xiong, Cao, and Dou (2021) introduced a Markov Chain-based method, which computes the element-wise multiplication between the integrated gradient matrix and the state transition matrix. Moreover, Xu, Yan, Ding and Liu (2022) proposed a Rollout attribution method, which computes the importance scores of different heads in the multi-head attention by using the integrated gradients algorithm. Additionally, this method utilizes the Rollout algorithm (Abnar & Zuidema, 2020) to calculate a series of matrix multiplications on the visualizations from each layer.

2.4. Other explanation method

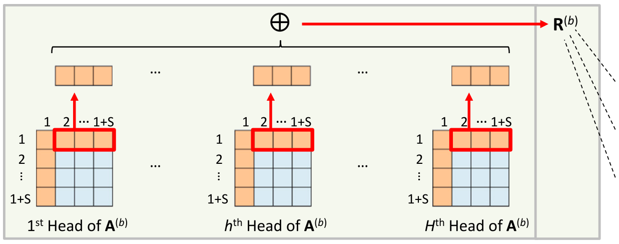
Different from the above types of methods, which are input-specific, Ghiasi et al. (2022) employed an activation maximization technique, which is input-agnostic, to visualize Transformer models. To enhance the visualization quality, various constraints are utilized, including total variance regularization, Gaussian smoothing constraint, color jitter augmentation, and color drift augmentation. Xie, Li, Cao, and Zhang (2023) proposed Vit-CX to cluster the feature maps from the attention module as the masks and perform multiple feedforwards to compute the impact scores according to the masked images, which are incorporated with random noise to mitigate artifacts. Vilas, Schaumlöffel, and Roig (2023) revealed that the intermediate representations in Transformers can be projected to the class embedding space by linear transformations and their back-propagations, which can further detect the class importance.

Among the visual explanation methods for Transformer networks, attention-based methods utilize attention weights that incorporate tangled semantics from multiple classes, limiting their focus to foreground regions where potential targets may exist. Some methods derived from the explanation of CNNs, such as relevance-based methods and gradient-based methods, do not fully take into account the unique characteristics of Transformer networks, resulting in unsatisfactory explanation results. Our work tries to tackle these issues through the following investigation.



V(b)
K(b)
Q(b)
O (b)
softmax
A (b)
Z (b)
$\frac{\partial y_c}{\partial Z^{(b)}}$
$\frac{\partial y_c}{\partial A^{(b)}}$
$\frac{\partial y_c}{\partial O^{(b)}}$
$\frac{\partial y_c}{\partial Q^{(b)}}$

Fig. 2. Diagram of the self-attention module and the gradient back-propagated to the query tensor $\mathbf{Q}^{(b)}$.



⊕
$\mathbf{R}^{(b)}$
y_1
y_c
y_C
1 2 ··· 1+S
1
2
1+S
···
1st Head of $\mathbf{A}^{(b)}$
hth Head of $\mathbf{A}^{(b)}$
Hth Head of $\mathbf{A}^{(b)}$

Fig. 3. Diagram of the extraction process for the attention relevance vector $\mathbf{R}^{(b)}$. $\mathbf{R}^{(b)}$ is a general explanations without distinguishing different categories of predictions. The dashed lines in the diagram represent corresponding relationships.

3. Method

In this section, we first analyze the semantic coupling problem in the self-attention module in the Transformer network. Then, we propose a gradient-decoupling-based relevance method for the visual explanation of Transformer. Specifically, the gradient-based class relevance computation, the attention aggregation and propagation, and the multi-head relevance integration are elaborated.

3.1. Problem analysis

A vision Transformer model typically contains multiple (assumed to be B) Transformer encoder blocks. Each of these encoder blocks consists of a multi-head self-attention module, an MLP module, a skip-connection layer, and normalization layers. In the following, we focus on the multi-head self-attention module in the Transformer encoder block.

First, we introduce the background knowledge and mathematical definitions of self-attention to facilitate the subsequent analysis and description. For the multi-head self-attention module (MHSA) in the b-th Transformer encoder block, its input tokens are projected into three tensors, i.e., the query tensor $\mathbf{Q}^{(b)} \in \mathbb{R}^{H\times(1+S)\times D}$, the key tensor $\mathbf{K}^{(b)} \in \mathbb{R}^{H\times(1+S)\times D}$ and the value tensor $\mathbf{V}^{(b)} \in \mathbb{R}^{H\times(1+S)\times D}$, where H denotes the number of heads, $(1+S)$ denotes the total length of class and spatial tokens (S corresponds to the length of spatial tokens), and D denotes the number of channels. As illustrated in Fig. 2, the attention weight matrix $\mathbf{A}^{(b)} \in \mathbb{R}^{H\times(1+S)\times(1+S)}$ is calculated by the scaled dot product (Vaswani et al., 2017) of the query tensor and the key tensor, which is written as follows:

$$\mathbf{A}^{(b)} = \mathrm{softmax}(\frac{\mathbf{Q}^{(b)}{\mathbf{K}^{(b)}}^{T}}{\sqrt{D}}). \tag{1}$$

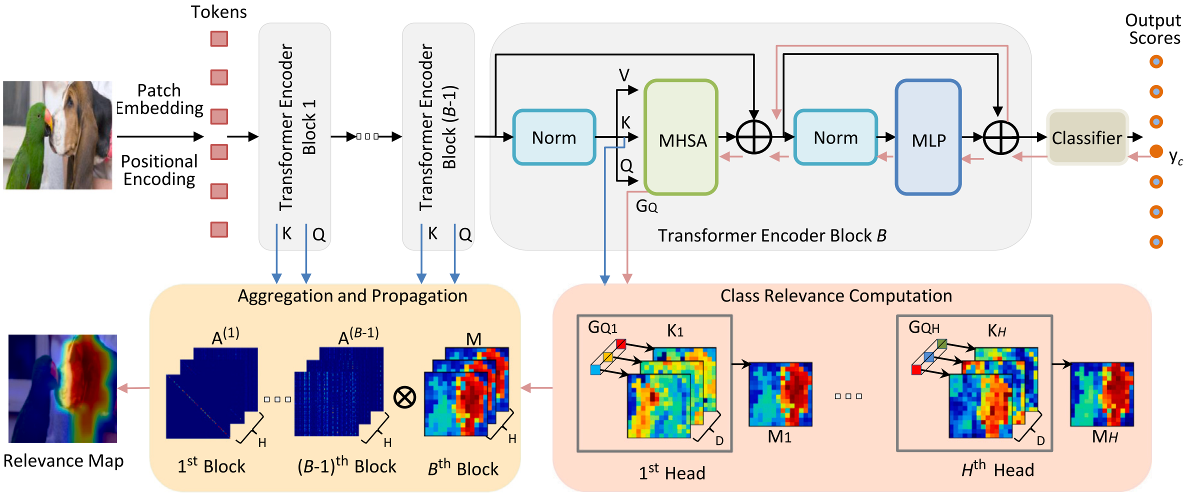


Fig. 4. Framework of the gradient-decoupling-based token relevance method (i.e., GradToken). "MHSA" denotes the multi-head self-attention module. On the lower right, the gradient $\mathbf{G}_Q$ is used to calculate the similarity with the key tensor $\mathbf{K}$ to obtain the class relevance $\mathbf{M}$. On the lower left, the relevance map $\mathbf{M}$ is propagated with the aggregated attention matrix $\mathbf{A}$. The pink lines denote the back-propagation. The blue lines denote feature transfer. Notice that K and M are converted into two-dimensional maps for the convenience of displaying.

The matrix multiplication between $\mathbf{Q}^{(b)}$ and $\mathbf{K}^{(b)}$ realizes the querying of the query tensor on the key tensor, obtaining the product between each pair of tokens within the tensors. Each row in $\mathbf{A}^{(b)}$ indicates the similarity of a single token to all tokens in that row. The attention weight matrix $\mathbf{A}^{(b)}$ is multiplied with the value tensor $\mathbf{V}^{(b)}$ to obtain the attention output $\mathbf{Z}^{(b)} \in \mathbb{R}^{H\times(1+S)\times D}$, as follows:

$$\mathbf{Z}^{(b)} = \mathbf{A}^{(b)}\mathbf{V}^{(b)}. \tag{2}$$

Then, we will analyze the self-attention matrix in explaining Transformer, which is adopted by our baseline method RawAtten (Clark et al., 2019). The first row in the attention weight matrix $\mathbf{A}^{(b)}$ obtained by Eq. (1) has a special meaning in that it reveals the similarity of the class token to each of the other (spatial) tokens. Therefore, the first row and the second column to $(1+S)$-th column in $\mathbf{A}^{(b)}$ can be taken out and averaged over multiple heads to obtain the attention relevance vector $\mathbf{R}^{(b)} \in \mathbb{R}^{S}$ (shown in Fig. 3):

$$\mathbf{R}^{(b)} = \frac{1}{H}\sum_{h=1}^{H} \mathbf{A}^{(b)}_{h,1,2:(1+S)}, \tag{3}$$

where the subscript of $\mathbf{A}^{(b)}_{h,1,2:(1+S)}$ corresponds to the head dimension and two spatial dimensions. The vector $\mathbf{R}^{(b)}$ can be reshaped into a two-dimensional map with the size of $N \times N$ (where $N = \sqrt{S}$) for visualization.

Since the class token in the attention output $\mathbf{Z}^{(b)}$ is trained and is relevant to the final prediction, the similarity corresponding to the first row in $\mathbf{A}^{(b)}$ contains certain semantics so that $\mathbf{R}^{(b)}$ can be used to highlight foreground regions. However, using $\mathbf{R}^{(b)}$ to explain the predictions of the Transformer model is not reliable. This is because $\mathbf{A}^{(b)}_{h,1,2:(1+S)}$ in Eq. (3) is computed from the class token $\mathbf{Q}^{(b)}_{h,1}$ in the query tensor with the key tensor. Besides, $\mathbf{Q}^{(b)}_{h,1}$ has not directly been supervised by the specific category information during the training procedure, resulting in its incapacity for distinguishing the semantics of different categories. Thus, $\mathbf{R}^{(b)}$ can only perceive the tangled semantics, rather than the specific semantics corresponding to a target category during the explanation.

3.2. Gradient-based class relevance computation

In order to solve the tangled semantic problems of RawAtten, we propose a gradient-decoupling-based method for computing the relevance of tokens, referred to as GradToken. GradToken uses gradients to decouple the class token so as to correspond to different semantic categories, as shown in Fig. 4. Specifically, the score $\mathbf{y}_c$ of the target category in the output layer is first selected. Then, following the chain rule, we can compute the gradient $\partial\mathbf{y}_c/\partial\mathbf{Z}^{(b)}$ of the target score $\mathbf{y}_c$ w.r.t. the multi-head self-attention $\mathbf{Z}^{(b)}$, by sequentially calculating its gradients w.r.t. MLP module, normalization layer, and skip connection layer. As shown in Fig. 2, the gradient of the target score $\mathbf{y}_c$ w.r.t. the attention weight matrix $\mathbf{A}^{(b)}$ is solved according to matrix multiplication as follows:

$$\frac{\partial\mathbf{y}_c}{\partial\mathbf{A}^{(b)}} = \frac{\partial\mathbf{y}_c}{\partial\mathbf{Z}^{(b)}}\mathbf{V}^{(b)^T}. \tag{4}$$

Denoting $\mathbf{Q}^{(b)}\mathbf{K}^{(b)^T}/\sqrt{D}$ in Eq. (1) by $\mathbf{O}^{(b)}$, the gradient of $\mathbf{y}_c$ w.r.t. $\mathbf{O}^{(b)}$ is:

$$\begin{aligned}\frac{\partial\mathbf{y}_c}{\partial\mathbf{O}^{(b)}} &= \frac{\partial\mathbf{y}_c}{\partial\mathbf{A}^{(b)}}\mathrm{softmax}^{-1}(\mathbf{O}^{(b)}) \\ &= \frac{\partial\mathbf{y}_c}{\partial\mathbf{Z}^{(b)}}\mathbf{V}^{(b)^T}\mathrm{softmax}^{-1}(\mathbf{O}^{(b)}),\end{aligned} \tag{5}$$

where $\mathrm{softmax}^{-1}(\cdot)$ denotes the derivative function of the $\mathrm{softmax}(\cdot)$. Subsequently, the gradient of the target score w.r.t. $\mathbf{Q}^{(b)}$ is derived according to the relationship between $\mathbf{Q}^{(b)}$ and $\mathbf{O}^{(b)}$ as follows:

$$\begin{aligned}\frac{\partial\mathbf{y}_c}{\partial\mathbf{Q}^{(b)}} &= \frac{1}{\sqrt{D}}\frac{\partial\mathbf{y}_c}{\partial\mathbf{O}^{(b)}}\mathbf{K}^{(b)} \\ &= \frac{1}{\sqrt{D}}\frac{\partial\mathbf{y}_c}{\partial\mathbf{Z}^{(b)}}\mathbf{V}^{(b)^T}\mathrm{softmax}^{-1}(\mathbf{O}^{(b)})\mathbf{K}^{(b)}.\end{aligned} \tag{6}$$

After the above gradient computation, the semantic information related to the selected target class is associated with the gradient of $\mathbf{Q}^{(b)}$. By choosing different output targets c, we can obtain $\partial\mathbf{y}_c/\partial\mathbf{Q}^{(b)}$ for different semantic classes.

Eq. (6) accomplishes the semantic decoupling for different classes. However, since the output of the last layer only contains the class token and not all tokens in the query tensor $\mathbf{Q}^{(B)}$ are trained, it is necessary to extract useful information from the gradient of $\mathbf{Q}^{(B)}$. As shown in Fig. 5, we extract the vector with semantic meaning (i.e., the gradient of the class token $\mathbf{G}^{(c)}_{Q_h} \in \mathbb{R}^{D}$) from the gradient $\partial\mathbf{y}_c/\partial\mathbf{Q}^{(b)}|_{b=B}$ obtained from Eq. (6):

$$\mathbf{G}^{(c)}_{Q_h} = \frac{\partial\mathbf{y}_c}{\partial\mathbf{Q}^{(B)}}[h,1,:], \tag{7}$$

where $[h,1,:]$ means that the h-th head, the first row (i.e., the class token), and all columns (i.e., all channels) are extracted. Thus, the $\mathbf{G}^{(c)}_{Q_h}$ obtained from Eq. (7) corresponds to the gradient of the class token in the h-th head of $\mathbf{Q}^{(B)}$ in the last layer.

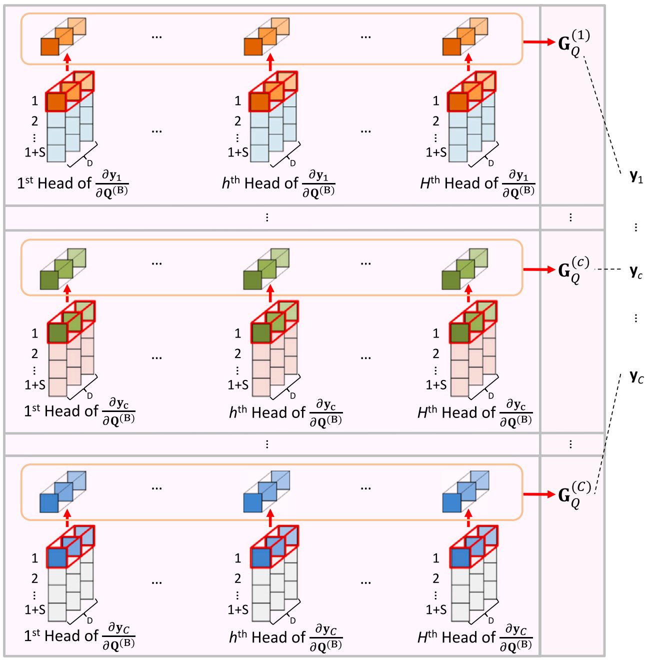

$\mathbf{G}_Q^{(1)}$
1
2
1+S
D
1st Head of $\frac{\partial y_1}{\partial Q^{(B)}}$
hth Head of $\frac{\partial y_1}{\partial Q^{(B)}}$
Hth Head of $\frac{\partial y_1}{\partial Q^{(B)}}$
$\mathbf{y}_1$
$\mathbf{G}_Q^{(c)}$
$\mathbf{y}_c$
1st Head of $\frac{\partial y_c}{\partial Q^{(B)}}$
hth Head of $\frac{\partial y_c}{\partial Q^{(B)}}$
Hth Head of $\frac{\partial y_c}{\partial Q^{(B)}}$
$\mathbf{y}_C$
$\mathbf{G}_Q^{(C)}$
1st Head of $\frac{\partial y_C}{\partial Q^{(B)}}$
hth Head of $\frac{\partial y_C}{\partial Q^{(B)}}$
Hth Head of $\frac{\partial y_C}{\partial Q^{(B)}}$

Fig. 5. Diagram of the extraction process for the gradient of the class token $\mathbf{G}_Q$. $\mathbf{G}_Q^{(c)}$ corresponds to the specific target category of prediction $\mathbf{y}_c$. The dashed lines in the diagram represent corresponding relationships.

Considering that the key tensor $\mathbf{K}^{(B)}$ is also trained, the proposed GradToken incorporate the information from the key tensor $\mathbf{K}^{(B)}$ for visualization. Similar to the conventional attention calculation, the decoupled gradient $\mathbf{G}_{Q_h}^{(c)}$ is used to calculate the similarity with $\mathbf{K}^{(B)}$, which can be achieved through a 1×1 convolution $\mathrm{Conv}(\cdot, \cdot)$, as follows:

$$\mathbf{M}_h^{(c)} = \mathrm{Conv}(\mathbf{K}_h, \mathbf{G}_{Q_h}^{(c)}), \quad h = 1, 2, \ldots, H, \tag{8}$$

where $\mathbf{M}_h^{(c)} \in \mathbb{R}^S$ is the class relevance vector of the h-th head for the cth target. $\mathbf{K}_h$ is the input variable of the convolution, denoting the matrix of the h-th head of the key tensor $\mathbf{K}^{(B)}$, and $\mathbf{G}_{Q_h}^{(c)}$ is utilized as the weight of the convolution. It is worth noting that the convolution weight here is not obtained by learning, but by computing the gradient of the target score $\mathbf{y}_c$ w.r.t. $\mathbf{Q}^{(B)}$ during the inference procedure.

In Eq. (8), $\mathbf{K}_h$ can be understood as high-level semantic features containing both foreground and background information. $\mathbf{G}_{Q_h}^{(c)}$ represents the gradient-decoupled class token, which is associated with the selected target class. By convolving $\mathbf{K}_h$ with $\mathbf{G}_{Q_h}^{(c)}$, we obtain the relevance between each spatial token and the selected target class. This reveals the importance of each spatial token to the target class in the prediction of the Transformer network.

3.3. Attention aggregation and propagation

The class relevance vector $\mathbf{M}_h^{(c)}$ obtained by Eq. (8) represents the visualization result for the B-th Transformer encoder block (i.e. the last encoder block before the classifier). To enhance the visualization results, we now consider how to visualize the relevance map of lower-layer encoder blocks. The existing method TransAttrib (Chefer et al., 2021b) also computes the relevance maps of attention weight matrices in the lower layers of the Transformer network. These maps are then multiplied together using the Rollout algorithm (Abnar & Zuidema, 2020) across all layers to obtain the final visualization result. However, it has been observed that the relevance maps obtained in the lower layers are inferior. This observation is similar to the findings of the previous investigations (Cheng, Fang, Liang, Zhang, Shen, & Wang, 2022; Selvaraju et al., 2020) on the visualization of CNNs. The lower layers of the network exhibit poorer semantic discriminability compared to the higher layers. During the visualization process for the lower layers, the back-propagated gradients tend to diverge, resulting in a loss of focus on the target.

However, the lower Transformer encoder blocks also contain rich spatial information that can be utilized to enhance the visualization results. Specifically, the weight matrix $\mathbf{A}^{(b)}$ from the lower encoder block implies the relevance between different spatial tokens and can be used to propagate the relevance from higher layers to lower layers. According to Eq. (1), the attention weight matrix $\mathbf{A}^{(b)}$ can be computed from the query tensor $\mathbf{Q}^{(b)}$ and the key tensor $\mathbf{K}^{(b)}$ in the Transformer encoder blocks. The multi-layer attention weight matrices $\mathbf{A}^{(b)}$ can be integrated through various aggregation and propagation schemes.

Four types of aggregation and propagation schemes are given as follows:

3.3.1. Rollout & element-wise multiplication

First, the attention weight matrices $\mathbf{A}_h^{(b)}$ of the lower layers are aggregated using the Rollout algorithm to obtain the aggregated matrix $\dot{\mathbf{L}} \in \mathbb{R}^{(1+S)\times(1+S)}$, as shown in the following equation:

$$\mathbf{L}^{(b)} = \mathbf{I} + \frac{1}{H}\sum_{h=1}^{H} \mathbf{A}_h^{(b)}, \quad b = 1, 2, \ldots, B-1, \tag{9}$$

$$\dot{\mathbf{L}} = \mathbf{L}^{(B-1)} \ldots \mathbf{L}^{(2)}\mathbf{L}^{(1)}, \tag{10}$$

where $\mathbf{I}$ denotes the identity matrix with the same dimension of $\mathbf{A}_h^{(b)}$. Then, we propagate the class relevance from the last layer to the lower layer by the element-wise multiplication:

$$\hat{\mathbf{M}}_h^{(c)} = \dot{\mathbf{L}}_{1,2:(1+S)} \odot \mathbf{M}_h^{(c)}, \quad h = 1, 2, \ldots, H, \tag{11}$$

where $\odot$ denotes the element-wise multiplication. $\dot{\mathbf{L}}_{1,2:(1+S)}$ indicates the similarity between the class token and the spatial tokens.

3.3.2. Rollout & matrix multiplication

First, Eq. (9) is used to aggregate the multi-head self-attention weight matrix from each layer, to obtain the matrix $\mathbf{L}^{(b)}$ (where $b = 1, 2, \ldots, B-1$). Then, the matrices $\mathbf{L}^{(b)}$ for all layers are multiplied together to obtain the aggregated matrix $\tilde{\mathbf{L}} \in \mathbb{R}^{(1+S)\times(1+S)}$:

$$\tilde{\mathbf{L}} = \mathbf{L}^{(1)}\mathbf{L}^{(2)} \ldots \mathbf{L}^{(B-1)}. \tag{12}$$

Finally, the class relevance from the last layer is propagated to the lower layers by matrix multiplication:

$$\hat{\mathbf{M}}_h^{(c)} = \tilde{\mathbf{L}}_{2:(1+S),2:(1+S)}\mathbf{M}_h^{(c)}, \quad h = 1, 2, \ldots, H, \tag{13}$$

where $\tilde{\mathbf{L}}_{2:(1+S),2:(1+S)} \in \mathbb{R}^{S\times S}$ denotes the submatrix corresponding to the spatial tokens extracted from $\tilde{\mathbf{L}}$. $\tilde{\mathbf{L}}_{2:(1+S),2:(1+S)}$ indicates the similarity between each pair of spatial tokens. It is worth noting that the order of matrix multiplication in Eq. (12) is different from that in Eq. (10) because they apply to different ways of relevance propagation.

3.3.3. Average & element-wise multiplication

First, the attention weight matrices $\mathbf{A}_h^{(b)}$ of the lower layers are aggregated by averaging operation to obtain the aggregated matrix $\bar{\mathbf{A}} \in \mathbb{R}^{(1+S)\times(1+S)}$, as follows:

$$\bar{\mathbf{A}} = \frac{1}{B}\frac{1}{H}\sum_{b=1}^{B-1}\sum_{h=1}^{H} \mathbf{A}_h^{(b)}. \tag{14}$$

Then, the class relevance from the last layer is propagated to the lower layers by element-wise multiplication:

$$\hat{\mathbf{M}}_h^{(c)} = \bar{\mathbf{A}}_{1,2:(1+S)} \odot \mathbf{M}_h^{(c)}, \quad h = 1, 2, \ldots, H. \tag{15}$$

3.3.4. Average & matrix multiplication

First, the attention weight matrices $\mathbf{A}_h^{(b)}$ of the lower layers are aggregated by averaging operation, calculated in the same way as in Eq. (14). Then, the class relevance from the last layer is propagated to the lower layers by matrix multiplication:

$$\hat{\mathbf{M}}_h^{(c)} = \bar{\mathbf{A}}_{2:(1+S),2:(1+S)}\mathbf{M}_h^{(c)}, \quad h = 1, 2, \dots, H. \tag{16}$$

Among the four aggregation and propagation schemes mentioned above, the aggregated matrix, i.e., $\dot{\mathbf{L}}_{1,2:(1+S)}$, $\tilde{\mathbf{L}}_{2:(1+S),2:(1+S)}$, $\bar{\mathbf{A}}_{1,2:(1+S)}$, and $\bar{\mathbf{A}}_{2:(1+S),2:(1+S)}$, are all related to spatial positions. However, the spatial tokens in the attention weight matrix of the last layer, i.e., $\mathbf{A}^{(B)}_{h,2:(1+S),2:(1+S)}$, has not been trained, so it is discarded during the aggregation and propagation process. Through experimental comparison (see Section 4.4.1), it shows that the fourth scheme (i.e., Average & Matrix Multiplication) yields the best results. In the fourth scheme, the averaging operation balances the influence of attention matrices from different layers, while the matrix multiplication operation adjusts the initial class relevance vector by weighted reorganization, which is reasonable. Therefore, the scheme of Average & Matrix Multiplication is chosen as the attention aggregation and propagation approach.

3.4. Multi-head relevance integration

The class relevance vector obtained by the above aggregation and propagation process has H heads. However, the final visualization map does not contain the multi-head dimension. Therefore, further integration of multi-head class relevance vector is required. The integration of multi-head class relevance vector in this section is not a simple post-processing step. It is important for the selection and fusion of multi-head semantics. Different integration schemes will lead to large differences in the generated visualization results. Two specific schemes of multi-head relevance integration are given as follows:

3.4.1. Average

The direct "average" scheme for multi-head integration is computationally simple, which computes the average values of the class relevance vector $\hat{\mathbf{M}}_h^{(c)}$ across the heads, as follows:

$$\hat{\mathbf{M}}^{(c)} = \frac{1}{H}\sum_{h=1}^{H}\hat{\mathbf{M}}_h^{(c)}. \tag{17}$$

However, there is a limitation in Eq. (17) for the multi-head relevance integration. It may amplify the influence of certain class relevance heads with negative values, leading to negative relevance in the generated visualization map. The qualitative and quantitative results in the experimental section provide evidence for this issue.

3.4.2. ReLU & average

To address the limitation of the above scheme, we first take positive values from each head and then averages them over multiple heads by the following formula:

$$\hat{\mathbf{M}}^{(c)} = \frac{1}{H}\sum_{h=1}^{H}\mathrm{ReLU}(\hat{\mathbf{M}}_h^{(c)}). \tag{18}$$

where the $\mathrm{ReLU}(\cdot)$ function is used to truncate the negative values to zero for each element in the vector. Eq. (18) only takes the positive value of each head of the class relevance vector, thus mitigating the influence of negative values from certain heads on the overall integration. This allows the generated visualization map to focus on the target region. Based on the experimental validation, the scheme of ReLU & Average achieves better results and is chosen as the multi-head relevance integration approach.

3.5. Method summary

As illustrated in Algotithm 1, the class relevance vector is obtained by Eqs. (6), (7) and (8) in Section 3.2. Then, the class relevance vector is propagated to lower layers by Eqs. (14) and (16) in Section 3.3. Finally, the multi-head relevance vector is integrated by using Eq. (18) in Section 3.4. The integrated relevance vector can be converted to a two-dimensional relevance map and then interpolated to the same size as the input image to generate a visualization of the Transformer model's prediction regarding the c-th category. Through the computations in the above formulations, we can achieve the disentangled class relevance, which is further enhanced by the propagation and integration operations, leading to better explanation results.

Algorithm 1: Method summary of GradToken

```
Input  : I: Image; c: class
Output: $\mathbf{V}^{(c)}$: Visualization map
$\mathbf{Q}^{(b)}, \mathbf{K}_h, \mathbf{A}_h^{(b)}, \mathbf{y}_c$ ← Feedforward with I;
$\partial\mathbf{y}_c/\partial\mathbf{Q}^{(b)}$ ← Compute the gradient of $\mathbf{y}_c$ w.r.t. $\mathbf{Q}^{(b)}$ by Eq. (6);
$\mathbf{G}_{Q_h}^{(c)}$ ← Extract the gradient of the class token from $\partial\mathbf{y}_c/\partial\mathbf{Q}^{(b)}$ by
  Eq. (7);
$\mathbf{M}_{\mathbf{h}}^{(\mathbf{c})}$ ← Convolve the gradient of the class token $\mathbf{G}_{Q_h}^{(c)}$ with the key
  tensor $\mathbf{K}_h$ by Eq. (8);
$\bar{\mathbf{A}}$ ← Aggregate attention weight matrices $\mathbf{A}_h^{(b)}$ by Eq. (14);
$\hat{\mathbf{M}}_{\mathbf{h}}^{(\mathbf{c})}$ ← Propagate the class relevance $\mathbf{M}_{\mathbf{h}}^{(\mathbf{c})}$ with $\bar{\mathbf{A}}$ by Eq. (16);
$\hat{\mathbf{M}}^{(\mathbf{c})}$ ← Integrate the multi-head relevance $\hat{\mathbf{M}}_{\mathbf{h}}^{(\mathbf{c})}$ by Eq. (18);
$\mathbf{V}^{(c)}$ ← Reshape and interpolate the integrated relevance $\hat{\mathbf{M}}^{(\mathbf{c})}$
```

Since other methods can also realize the visual explanation of Transformer to some extent, we discuss the connections and differences between the proposed GradToken and three other typical methods:

3.5.1. Rollout

Both Rollout and GradToken make use of multi-level attention weight matrices $\mathbf{A}$ for computation. However, the results obtained by Rollout are category-independent. The proposed GradToken uses gradients to decouple the semantics of different categories such that the target category can be distinguished from other categories.

3.5.2. TransAttrib

Both TransAttrib and GradToken make use of gradient information for computation. However, TransAttrib computes the gradients with respect to the attention weight matrix $\mathbf{A}$, while GradToken computes the gradients with respect to the query tensor $\mathbf{Q}$ and then convolves it with the key tensor $\mathbf{K}$. Furthermore, the gradients obtained through GradToken contain richer information (as seen in Eqs. (5) and (6)). Therefore, there are significant differences in the gradient computation processes between TransAttrib and GradToken.

3.5.3. GradCAM

Both GradCAM and GradToken can be seen as firstly computing gradients to obtain a weight vector, which is then used to weigh a multi-dimensional matrix. However, in GradCAM, when computing gradients with respect to the attention weight matrix $\mathbf{A}$, each attention head can only be assigned a single value, which is then used to weigh over the attention head. This process does not preserve channel information. In contrast, GradToken performs convolution independently for each attention head, where each convolution operation can capture information from multiple channels (as shown in Fig. 4). Thus, GradToken is capable of producing more accurate visualizations by leveraging channel information.

Through the above discussion, we can find that the proposed method has significant improvements over the other methods, especially in the aspects of the gradient and convolution computations for relevance, leading to better visual explanation.

4. Experiments

To evaluate the effectiveness of the proposed method, extensive quantitative and qualitative experiments are conducted in this section, including semantic segmentation experiments, perturbation experiments, generalization experiments (i.e., language evaluations), and visual comparison experiments. Then, ablation experiments are performed on aggregation and propagation of low-layer attention, integration of multi-head relevance, and depth of class relevance propagation. In the following experiments, the proposed GradToken is compared with ten other advanced methods, including GradCAM (Selvaraju et al., 2020), LRP (Bach et al., 2015), PLRP (Voita et al., 2019), RawAtten (Clark et al., 2019), Rollout (Abnar & Zuidema, 2020), TransAttrib (Chefer et al., 2021b), AttCAT (Qiang et al., 2022), ViT-CX (Xie et al., 2023), absLRP (Vukadin et al., 2024), and AGCAM (Leem & Seo, 2024).

4.1. Experimental settings

4.1.1. Evaluation datasets and models

In order to evaluate the effectiveness and reliability of the proposed method, we follow the settings of Chefer et al. (2021b) to conduct the experiments on four datasets: ImageNet 2012 object classification (Russakovsky et al., 2015), ImageNet-Segmentation (Guillaumin et al., 2014), PASCAL VOC (Everingham et al., 2010) and Movies Reviews (Zaidan & Eisner, 2008). The **ImageNet** dataset consists of approximately 1.2 million training images, 50,000 validation images, and a total of 1,000 categories. **ImageNet-Segmentation** is an improved dataset based on ImageNet, where binary segmentation masks are annotated on images. This dataset contains 4,276 images and a total of 445 categories. The **VOC** dataset contains 21 classes including the background class and three standard splits, i.e., training set (1464 images), validation set (1449 images) and test set (1456 images), and an extra augmented training split, i.e., trainaug set (10582 images) (Hariharan, Arbelaez, Bourdev, Maji, & Malik, 2011). The **Movies Reviews** dataset contains 1600 reviews in the training set, 200 reviews in the validation set, and 200 reviews in the test set. The experiments are conducted on the classical Transformer models, i.e., ViT-B/16 (Dosovitskiy et al., 2021) and BERT-B (Devlin, Chang, Lee, & Toutanova, 2019), for the vision task and the language task, respectively.

4.1.2. Evaluation metrics

The segmentation experiments adopt a referenced evaluation metric, where ground truth labels are provided as references. In the experiments, the target localization accuracy is evaluated by comparing the visualization result with the segmentation label. Specifically, three segmentation evaluation metrics are adopted: pixel accuracy (PAcc), mean average precision (mAP), and mean intersection over union (mIoU).

The perturbation (Chefer et al., 2021b) experiments adopt a weak reference evaluation metric, which provides the original classification labels for evaluation of the reliability of the explanatory results. This evaluation metric is not affected by manually set thresholds. Given a generated visualization, the pixels of the input image are gradually erased based on the sort order of the response strength in the visualization. The average top-1 accuracy of the classification is recorded for each step of pixel perturbation. Then, the perturbation score is measured by calculating the area under the accuracy curve (AUC) based on the recorded multi-step accuracy results. In the perturbation experiments, 10% of the image pixels are erased in each step.

For the language evaluation, we follow the setting of ERASER (DeYoung et al., 2020) and TransAttrib (Chefer et al., 2021b) to validate if the generated explanatory results support predictions of the sentiment classification. In particular, the token-F1 score on the test set is adopted to evaluate the results generated by competitors. The token-F1 score measures the overlap between the human-labeled rationale tokens and top-k tokens generated by explanation methods, where $k \in [10, 80]$ with steps of 10 tokens.

Table 1
Segmentation performance (%) of the proposed GradToken and competitors on the ImageNet-Segmentation and VOC validation datasets. Results of Rollout, RawAtten, GradCAM, LRP, PLRP, and TransAttrib on the ImageNet-Segmentation dataset are from the literature (Chefer et al., 2021b). Higher values in the table indicate better results. Bold and underline denote the best and second-best results under each evaluation criterion. ViT-CX needs multiple feedforward passes. Other methods need a single feedforward or backward pass..

Method	ImageNet-Segmentation			VOC	
	PAcc	mAP	mIoU	PAcc	mIoU
Rollout (ACL2020)	73.54	84.76	55.42	64.15	24.85
RawAtten (ACLW2019)	67.84	80.24	46.37	61.21	18.40
GradCAM (IJCV2020)	64.44	71.60	40.82	69.03	24.85
LRP (PLOS ONE2015)	51.09	55.68	32.89	46.79	16.45
PLRP (ACLW2019)	76.31	84.67	57.94	66.76	29.89
TransAttrib (CVPR2021)	79.70	86.03	61.95	76.43	44.31
AttCAT (NeurIPS2022)	71.78	74.62	47.54	70.05	24.36
ViT-CX (IJCAI2023)	76.95	75.03	56.71	74.24	37.88
absLRP (ACM TIST2024)	71.04	78.37	52.26	48.75	29.90
AGCAM (2024)	81.15	**87.77**	63.70	78.73	42.47
GradToken (Ours)	**84.51**	86.10	**68.24**	**79.48**	**46.02**

4.2. Quantitative evaluation

To evaluate the localization accuracy, explanatory reliability, and generalization of the proposed method, we perform segmentation experiments, perturbation experiments, and generalization experiments, respectively in the following.

4.2.1. Segmentation experiment

Ideally, a visual explanation for model classification decisions should focus on the region where the target is located. In other words, if the visualization result generated by the explanatory method can successfully locate the target, it indicates that the method has a good visual explanatory effect. Therefore, to validate the effectiveness of the proposed explanation method, we conduct the segmentation experiments by evaluating the target localization accuracy.

First, based on the predictions of the Transformer model, the class with the highest classification score is selected as the target for visual explanation on the ImageNet-Segmentation dataset. On the VOC dataset, the ground truth class labels are chosen as the targets. Then, the explanation methods are adopted to generate visualization maps. Note that the post-processing of normalization and binarization are applied to all methods to obtain segmentation maps from visualization maps. We search the best thresholds with the step of 0.1 within the range of [0, 0.9] after normalization for each method to achieve the best performance.

Table 1 presents the segmentation results obtained by the proposed GradToken and ten other methods on the ImageNet-Segmentation and VOC validation datasets. Compared to the other methods, the proposed GradToken achieves the highest PAcc and mIoU on both datasets. On the ImageNet-Segmentation dataset, in terms of PAcc, the proposed GradToken achieves a score of 84.51%, which is 16.67% higher than the baseline method RawAtten (67.84%). This demonstrates that the proposed GradToken effectively focuses on the target of explanation by decoupling the semantic information of class token using gradients. In terms of mIoU, the proposed GradToken achieves a score of 68.24%, which is 27.42% higher than the classic CNN explanation method GradCAM (40.82%). This is mainly because the proposed GradToken computes not only the dimensions of attention heads but also the dimensions of attention channels. Furthermore, PAcc and mIoU obtained by GradToken are respectively 3.36% and 4.54% higher than the second-best method AGCAM (i.e., 81.15% and 63.70%). mAP obtained by GradToken is 7.73% higher than absLRP, but 1.67% lower than AGCAM. This might be due to AGCAM using a special normalization on attention maps (Leem & Seo, 2024).

Multi-class semantic segmentation experiment on the VOC dataset is more challenging for all the explanation methods, considering none of these methods has an mIoU of more than 50%. Though, GradToken achieves an mIoU of 46.02%, which is 1.71% higher than TransAttrib and 16.12% higher than absLRP. Overall, the proposed GradToken exhibits the outstanding segmentation performance on the ImageNet-Segmentation and VOC datasets, indicating GradToken possesses better accuracy and target selectivity for visual explanation of Transformer models.

4.2.2. Perturbation experiment

In the perturbation experiments, the model's predictions are measured. This is a straightforward way to validate if the generated explanations are accountable for the model's predictions. Thus, to further validate the reliability of the proposed method, we conduct perturbation experiments, including positive and negative ones.

For positive perturbation experiments, the image pixels are erased in the decreasing order of response strength. In an ideal visualization result, the strongest response should have the greatest impact on the classification prediction. Hence, when it is erased, the accuracy should drop sharply, resulting in a smaller AUC. For negative perturbation experiments, the image pixels are erased in the increasing order. Erasing unimportant pixels leads to a slow decrease in accuracy, resulting in a higher AUC. We set the target category for explanation in two ways. One way is the "label unknown" setting, where no class label is provided in advance, and the class with the highest predicted score is selected as the target category for explanation. The other way is the "label known" setting, where the class label in the dataset is used as the target category for explanation.

Table 2
AUC scores (%) obtained by the proposed GradToken and competitors in the perturbation experiments on the ImageNet validation set. The results of competitors are from the literature (Chefer et al., 2021b). ↓ denotes the lower the better. ↑ denotes the higher the better. Bold and underline denote the best and second-best results under each evaluation criterion.

Method	Positive ↓		Negative ↑	
	Label Unknown	Label Known	Label Unknown	Label Known
Rollout (ACL2020)	20.05	–	53.1	–
RawAtten (ACLW2019)	23.99	–	45.55	–
GradCAM (IJCV2020)	34.06	33.56	41.52	42.02
LRP (PLOS ONE2015)	41.94	41.93	43.49	43.49
PLRP (ACLW2019)	19.64	19.64	50.49	50.49
TransAttrib (CVPR2021)	17.03	16.04	54.16	55.04
AttCAT (NeurIPS2022)	21.00	20.02	39.54	41.69
ViT-CX (IJCAI2023)	**14.78**	**13.72**	<u>55.86</u>	<u>57.57</u>
GradToken (Ours)	<u>15.97</u>	<u>14.89</u>	**58.37**	**59.61**

Table 2 presents the AUC scores of the proposed GradToken and eight other methods on the ImageNet validation set for two ways (i.e., a total of four types) of perturbation experiments. Rollout and RawAtten only use the attention weight matrices in the Transformer network when generating the visualization maps and do not consider the relationship between the attention matrices and the target in the output. Thus, the experimental results for these two methods with label known setting are omitted, as the given labels are meaningless to them. It is also observed that the LRP and PLRP methods achieve nearly identical AUC scores for both ways of label unknown setting and label known setting. This is because these two methods cannot differentiate the target category during visualization, resulting in the same visualization results regardless of the selected target.

The proposed GradToken achieves AUC scores of 15.97% for the label unknown setting and 14.89% for the label known setting in the positive perturbation experiment (lower AUC is better). On the other hand, in the negative perturbation experiment (higher AUC is better), GradToken achieves 58.37% for the label unknown setting and 59.61% for the label known setting. Compared to the baseline RawAtten, the proposed GradToken decreases the positive perturbation AUC score by 8.02% and increases the negative perturbation AUC score by 12.82%. Compared to GradCAM, the proposed GradToken achieves significant performance improvements, i.e., improvements of 18.09% for the label unknown setting and 18.67% for the label known setting in the positive perturbation, and improvements of 16.85% for the label unknown setting and 17.59% for the label known setting in the negative perturbation. The proposed GradToken shows significant advantages in AUC scores compared to Rollout, RawAtten, GradCAM, LRP, and AttCAT. This indicates that the proposed method possesses stronger target selection capability, further validating its reliability. Compared to ViT-CX, GradToken achieves better performance in the negative perturbation experiment with the label unknown setting, and achieves competitive results in the positive perturbation experiments. It is understandable that ViT-CX has an advantage in the perturbation metric, since the method implementation of ViT-CX is iteratively evaluating the impact of mask on the predicted score.

In summary, the proposed GradToken achieves the superior performance in the perturbation experiments on the ImageNet validation set, demonstrating the strong reliability of the generated visualization results.

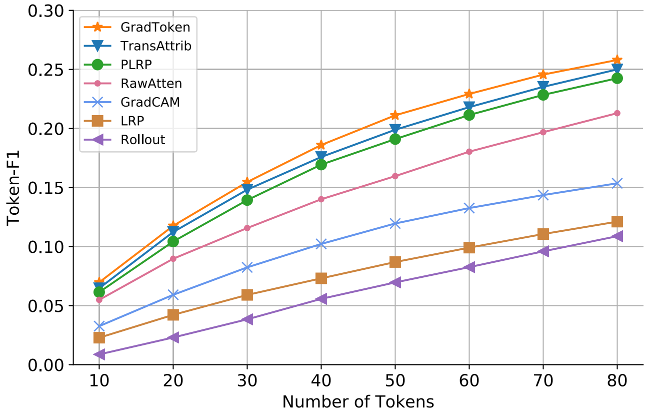


Fig. 6. Token-F1 scores obtained by the proposed GradToken and six other methods on the Movie Reviews dataset. This figure shows, as the top-k tokens ($k \in [10, 80]$) are selected at each step as the rationals from the explanatory results, the matching degree (token-F1) between these explanatory rationales and the ground truth rationales.

4.2.3. Generalization on the language model

To verify the generalization of the proposed method, we conduct the experiment on the language model. Specifically, we test on the BERT model (Devlin et al., 2019) which is trained on the Movie Reviews training set for the sentiment classification task. The input of the model is a paragraph of text, and the output is the classification of sentiment. The ground truth rationales are annotated by human workers to point out which textual segments (tokens) are important for sentiment classification. Explanation methods can assign importance scores to all input tokens, which are sorted according to the scores. Top-k tokens can be chosen as the rationals for the prediction corresponding to the setting of $k \in [10, 80]$. The matching degree between the rationals extracted by explanation methods and the ground truth rationales is evaluated with the token-F1 score on the Movie Reviews test set. A higher matching degree (i.e., token-F1 score) indicates a better explanation for the model's classification decision.

Fig. 6 shows token-F1 score obtained by each method corresponding to top-k tokens chosen as the rationals. Rollout and LRP achieve lower token-F1 scores compared to other methods, because these two methods lack target selectivity. Benefiting from the gradient decoupling for the target class, GradToken outperforms all competitors at each step, which indicates the explanatory results generated by GradToken are matched better with the labeled rationals than the competitors. Especially when the number of tokens increases, GradToken exhibits a more significant advantage in terms of the token-F1 scores over other methods, e.g., RawAttn, GradCAM, LRP, and Rollout. This experiment shows the superiority and the generalization ability of GradToken on the language model, except for the vision model.

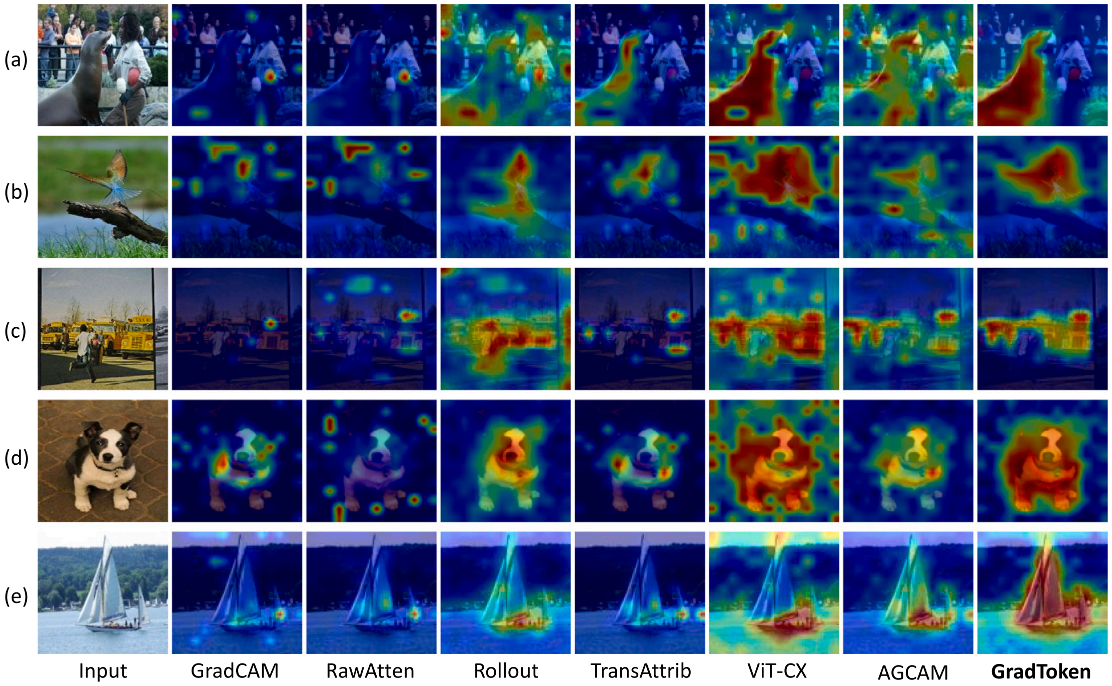


Fig. 7. Qualitative comparison of the single-class visualizations generated by the proposed GradToken and five other methods.

4.3. *Qualitative comparison*

To visually analyze and validate the proposed GradToken, we qualitatively compare GradToken with five competitors, including GradCAM (Selvaraju et al., 2020), ViT-CX (Xie et al., 2023), RawAtten (Clark et al., 2019), Rollout (Abnar & Zuidema, 2020), and TransAttrib (Chefer et al., 2021b), for both single-class and multi-class visual explanations of Transformer models.

4.3.1. *Single-class visualization*

After a Transformer model makes a prediction, the class with the highest prediction score is selected as the target for visual explanation. Fig. 7 shows the visualization results for the Transformer's prediction of the single-class target, where the samples are from the ImageNet dataset. The second to seventh columns display the generated visual heatmaps for the samples, where red indicates a higher relevance to the target and blue indicates a lower relevance.

As shown in Fig. 7, GradCAM and RawAtten focus on relatively small regions in most cases. Both methods also capture some irrelevant background areas. For example, in Fig. 7(b) and (d), which contains a bird and a dog respectively, GradCAM and RawAtten highlight some background regions. The proposed GradToken, benefiting from the utilization of gradients of the target, effectively highlights the target, with less background noise in the generated visualizations. Compared to RawAtten, Rollout and AGCAM can capture larger regions of interest. However, Rollout and AGCAM also struggle to differentiate between target objects and other non-target objects, as seen in the sea animal and people in Fig. 7(a), and the bird and wood in Fig. 7(b). ViT-CX hilights larger regions of objects while with more noise in the background, such as the cases in Fig. 7(b) and (d). In contrast, the proposed GradToken, by decoupling the class token, shows better target selectivity in multiple examples. Compared to Rollout, TransAttrib can more effectively focus on the target class. However, the visualizations generated by TransAttrib tend to focus on smaller regions, such as the small region of the sailboat in Fig. 7(e), and it also ignores most of the features of multiple buses in Fig. 7(c). On the other hand, the proposed GradToken benefits from the advantages of low-layer attention aggregation and propagation, allowing it to focus on larger target regions. This also demonstrates the global perception ability of the Transformer model.

Overall, compared to the other methods, the visualizations generated by GradToken can more accurately and completely focus on the regions where the target class is present.

4.3.2. *Multi-class visualization*

To validate the reliability of the proposed GradToken, we also test the visual explanations with regard to multi-class targets in images. A common way to examine the reliability of explanatory methods is to assess their sensitivity to different classes. Ideally, the explanatory results should differ for classes that have different semantics. If the results for different classes vary significantly, it indicates a certain extent of reliability in the explanatory method. Conversely, if the results for different classes are exactly the same, it suggests poor reliability of the explanatory method. Fig. 8 illustrates the visualization results of the proposed GradToken and five other methods for the Transformer's predictions of multi-class targets. The texts above and below the input images in the figure represent the target classes selected in the visual explanations.

In Fig. 8(a), the image of a cat and dog is a classic example commonly used in the field of explanations. The bodies of the cat and dog have similar colors, and the image contains variations in lighting and shadows, posing challenges to achieving an effective visual explanation. It can be observed that RawAtten mainly focuses on the dog, while Rollout focuses not only on the dog but also on the cat and background regions. However, these two methods fail to provide discriminative explanations for different target classes. GradCAM, ViT-CX and TransAttrib can discriminate different targets, but the regions they focus on are not precise enough. For example, GradCAM only focuses on the belt of the dog, which is a non-discriminative feature. In contrast, the proposed GradToken successfully highlights regions relevant to both the dog and cat classes. In Fig. 8(b), the image of elephants and zebras includes multiple instances, each of which is in small size in the view and is cluttered with each other. When explaining the elephant class, RawAtten fails to effectively highlight the region where the elephants are located. When explaining the zebra class, TransAttrib and GradCAM ignore the rightest zebra. However, the

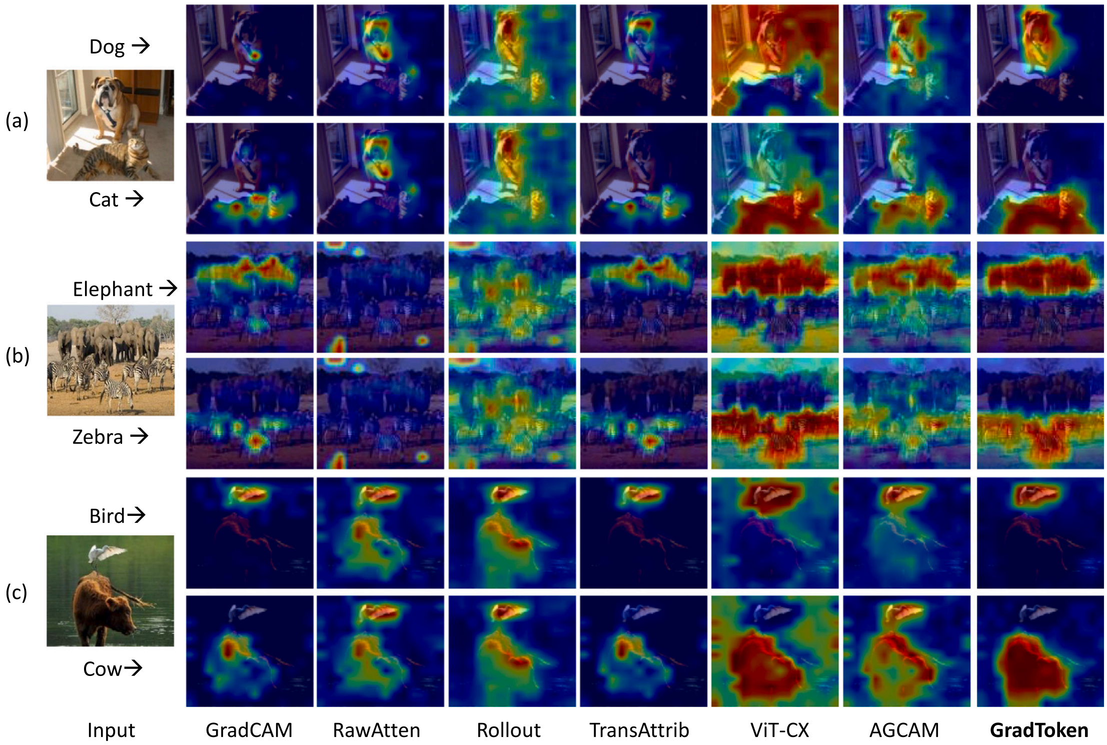


Fig. 8. Qualitative comparison of the multi-class visualizations generated by the proposed GradToken and five other methods. The texts above and below the input images indicate the target classes of the visualizations.

proposed GradToken can effectively highlight the multiple instances of elephants and zebras. In Fig. 8(c), a bird standing on a cow is a rare scene. RawAtten, Rollout, ViT-CX, and AGCAM struggle to distinguish between the target class and the non-target class or background. The visualizations generated by GradCAM and TransAttrib only highlight a small portion of the cow's back. On the other hand, the proposed GradToken can not only distinguish between the bird and the cow, but also accurately and completely highlight the regions relevant to the targets. Compared to the other methods, the visualizations generated by GradToken provide more reliable and accurate explanations for different target classes.

In summary, the comparative results of the single-class target visualization experiments (Fig. 7) and the multi-class target visualization experiments (Fig. 8) validate the accuracy and reliability of the proposed GradToken for visual explanations. These results also demonstrate the effectiveness of the designs including the gradient decoupling and the attention propagation. Furthermore, the qualitative comparative experiments complement the quantitative experiments and support their results from another perspective.

4.4. Ablation study

In this section, we conduct ablation studies on the ImageNet-Segmentation dataset to investigate the impact of attention aggregation and propagation, multi-head relevance integration, and depth of class relevance propagation on visualization results.

Third, Element-wise Multiplication contributes to mAP, while Matrix Multiplication contributes to PAcc. Comparing the aggregation and propagation schemes of GradToken-1 vs. GradToken-2, and GradToken-3 vs. GradToken-4, it can be observed that Element-wise Multiplication helps improve mAP (i.e. 88.01% >83.57% and 86.98% >86.10%). In contrast, Matrix Multiplication then helps to improve PAcc (i.e. 82.14% >81.73%, 84.51% >81.49%).

Table 3
Effect of using different aggregation and propagation schemes on segmentation performance (%). Higher values indicate better results. ⊙ denotes the element-wise multiplication. ⊗ denotes the matrix multiplication. Bold font denotes the optimal result under each evaluation criterion.

Method	Rollout	Average	⊙	⊗	PAcc	mAP	mIoU
GradToken-0	–	–	–	–	77.67	77.09	58.28
GradToken-1	✓	–	✓	–	81.73	**88.01**	65.43
GradToken-2	✓	–	–	✓	82.14	83.57	65.11
GradToken-3	–	✓	✓	–	81.49	86.98	63.99
GradToken-4	–	✓	–	✓	**84.51**	86.10	**68.24**

4.4.1. Effect of attention aggregation and propagation

To evaluate the influence of different attention aggregation and propagation schemes on the generation of visualization maps, we investigated four types of attention aggregation and propagation schemes in GradToken, along with a scheme without aggregation or propagation. They are : GradToken-0 (no aggregation or propagation); GradToken-1 (Rollout & Element-wise Multiplication); GradToken-2 (Rollout & Matrix Multiplication); GradToken-3 (Average & Element-wise Multiplication); GradToken-4 (Average & Matrix Multiplication).

Table 3 shows the segmentation results obtained by GradToken using the above five aggregation and propagation schemes on the ImageNet-Segmentation dataset. The experimental results indicate the followings:

First, each type of attention aggregation and propagation contributes to improved explanations. As shown in Table 3, GradToken-0 without aggregation and propagation achieves a PAcc of 77.67%, an mAP of 77.09%, and an mIoU of 58.28%. After applying attention aggregation and propagation, PAcc is improved to 84.51% (GradToken-4), mAP is improved to 88.01% (GradToken-1), and mIoU is improved to 68.24% (GradToken-4). Low-layer attention aggregation and propagation plays a significant role in improving all three scores.

Second, different aggregation and propagation schemes are suitable for different metrics. GradToken-1, which uses Rollout & Element-wise

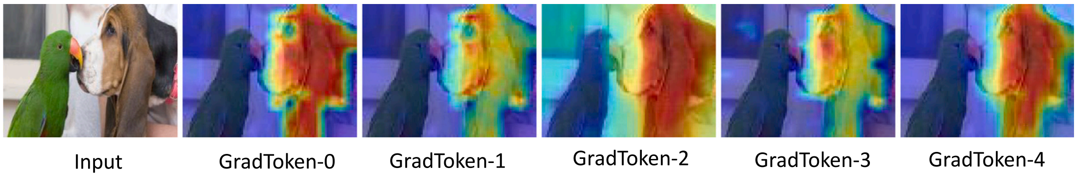


Fig. 9. Visualization maps generated by different aggregation and propagation schemes.

Table 4
Effect of using different multi-head relevance integration schemes on segmentation performance (%). Higher values indicate better results. Bold font denotes the optimal result under each evaluation criterion.

Method	average	ReLU & average	PAcc	mAP	mIoU
GradToken-i	✓	–	75.11	74.28	51.32
GradToken-ii	–	✓	**84.51**	**86.10**	**68.24**

Multiplication as the aggregation and propagation scheme, achieves the highest mAP (88.01%). GradToken-4, which uses Average & Matrix Multiplication as the aggregation and propagation scheme, achieves the highest PAcc and mIoU (84.51% and 68.24%, respectively). However, no single aggregation and propagation scheme performs best on all three metrics simultaneously.

In addition to the quantitative analysis, Fig. 9 presents qualitative comparison results. GradToken-0 without aggregation and propagation leads to generating holes in the middle of the target region, and missing key features like the dog's eyes. GradToken-1 highlights key features that are already present, but fails to compensate for the missing features. GradToken-2 can compensate for missing features in the hole positions but results in an excessively large target region, even generating responses in the background region. GradToken-3 highlights some background regions on the left, and introduces new holes in the dog's neck on the right. GradToken-4 not only compensates for missing key features in the hole positions but also avoids highlighting the background region. Visually, GradToken-4 with the aggregation and propagation scheme of Average & Matrix Multiplication exhibits the best performance. Considering the optimal performance achieved in terms of PAcc, mIoU, and visual results, Average & Matrix Multiplication is selected as the attention propagation and aggregation scheme in the proposed method.

4.4.2. Effect of multi-head relevance integration

To evaluate the influence of different multi-head relevance integration schemes on the visualizations generated by GradToken, we investigate two variants of multi-head relevance integration in GradToken: GradToken-i (Average) and GradToken-ii (ReLU & Average). Table 4 presents the segmentation results obtained by GradToken with the two multi-head relevance integration schemes on the ImageNet-Segmentation dataset, and Fig. 10 illustrates qualitative comparison results.

As shown in Table 4, directly averaging the class relevance vectors across multiple heads leads to lower PAcc, mAP, and mIoU. As depicted in Fig. 10, when visualizing the target class ("dog") using GradToken-i, which directly averages the class relevance vectors, the values of the target region turn negative. When the visualizations corresponding to the class relevance vectors from the 12 heads are displayed one by one (as shown in Fig. 10), it can be observed that the visualizations corresponding to the class relevance vectors from the second, fourth, seventh, eighth, ninth, and eleventh heads are negative. If the class relevance vectors of all heads are simply added together, some heads' negative values in the region of interest may exceed the sum of positive values from other heads. Therefore, as an improvement to the multi-head integration scheme in GradToken-i, GradToken-ii applies ReLU activation to all class relevance vectors and then computes their average, effectively suppressing the impact of negative values from certain heads on the overall visualization. As shown in the visualization result of GradToken-ii on the left side of Fig. 10, GradToken-ii successfully highlights the target region. From Table 4, it can also be observed that GradToken-ii, which adopts the ReLU & Average multi-head integration scheme, achieves an improvement of 9.40% in PAcc, an improvement of 11.82% in mAP, and an improvement of 16.82% in mIoU.

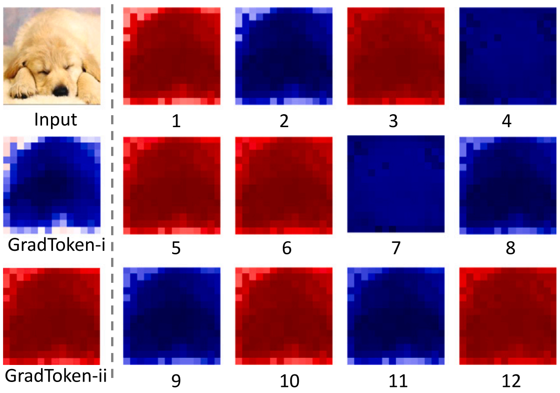


Fig. 10. Multi-head relevance integration and distribution of positive and negative values across different heads. The left side shows the input image and the visualization maps obtained by using two different multi-head integration schemes. The right side shows the visualization maps of twelve heads. The numbers in the figure indicate the head index. Red indicates the positive value and blue indicates the negative value. Some heads with negative values cause the integrated relevance of the target to turn negative in GradToken-i, while having no effect on GradToken-ii.

In summary, GradToken-ii, with the ReLU & Average integration scheme, effectively integrates the class relevance vectors from multiple heads, resulting in improved segmentation performance. Hence, ReLU & Average is selected as the multi-head relevance integration scheme.

4.4.3. Effect of class relevance propagation depths

To analyze the influence of the depth of class relevance propagation on the visualization results of GradToken, we investigate propagating the high-layer class relevance to different lower layers, specifically propagating it in reverse order to the eleventh, tenth, ..., second, and first layers of the network. Similarly, the segmentation results are used to evaluate the visualization performance with regard to the propagation depth.

As shown in Fig. 11, when class relevance is back-propagated from the last layer to the eleventh layer, all three scores (PAcc, mAP, and mIoU) decrease significantly compared to the case without propagation. As the depth of propagation increases, the three scores gradually improve. When class relevance is back-propagated to the eighth layer, the scores start to surpass those obtained without class relevance propagation and continue to increase in the subsequent propagations. Notably, when class relevance is back-propagated to the second layer, mAP and mIoU reach their highest values of 86.12% and 68.27%, respectively. However, when class relevance continues to be propagated to the first layer, mAP and mIoU slightly decrease, while PAcc reaches its highest value of 84.51%. This may be due to the fact that the attention weight matrix in the lowest layer emphasizes local

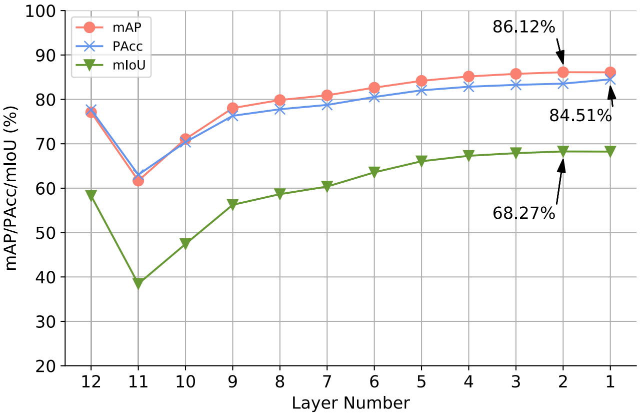


Fig. 11. The effect of class relevance back-propagation depths on segmentation performances. The highest score of each metric is annotated in the figure. Notice that the layer number "12" denotes no back-propagation.

spatial similarities, so propagation through the first layer leads to a slight reduction in the region of interest, thereby affecting mAP and mIoU. However, the enhancement of local similarities can highlight salient regions, thus improving PAcc. Taking into account the overall performance of PAcc, mAP, and mIoU, we select the lowest layer as the destination for class relevance propagation, i.e., back-propagating from the last layer to the first layer.

5. Conclusion

To address the interpretability of vision Transformer networks, we have proposed GradToken, a gradient-decoupling-based method, to compute the token relevance. By calculating the gradients of the output layer's target score with respect to the class token, GradToken decouples the tangled semantics within the class token and associates them with the semantics of different categories. Furthermore, GradToken performs convolution operations between the decoupled class token and spatial tokens, obtaining the relevance vector for each category, which can be transformed into the corresponding visualization map for Transformer's explanation. Experimental results on the ImageNet, ImageNet-Segmentation, and VOC datasets have shown that, compared to other state-of-the-art methods, GradToken not only provides discriminative explanations for different targets but also generates visualizations with more accurate target boundaries and less background noise, thus resulting in superior performance and reliable explanation.

Limitations. This work mainly focuses on the visual explanation for vision Transformer and explores a little on language Transformer. In future work, the explanation method can be expanded to other modal data, e.g., audio or text-image/video multimodalities. The main idea of gradient decoupling can be reused in the new tasks, but the target of the gradient computation should be adjusted corresponding to the input and output modality. Besides, the proposed method only addresses the explanation of Transformer networks with class tokens (e.g., ViT). To adapt to other Transformer architectures without class tokens, one possible solution is treating the average token of query tensors as the class token to compute the gradient and further obtain class-aware relevance.

CRediT authorship contribution statement

Lin Cheng: Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Yanjie Liang:** Writing – review & editing, Validation, Formal analysis. **Yang Lu:** Writing – review & editing, Supervision, Funding acquisition. **Yiu-ming Cheung:** Writing – review & editing, Validation.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used OpenAI's ChatGPT in order to check grammar and improve readability and language. The paper is original by the author, and the tool does not provide any ideas or semantic changes. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This study was supported in part by the National Natural Science Foundation of China under Grant 62376233; in part by the General Research Fund of Research Grants Council under Grant 12202622; in part by the Natural Science Foundation of Fujian Province under Grant 2024J09001; in part by the China Fundamental Research Funds for the Central Universities under Grant 20720230038; and in part by Xiaomi Young Talents Program.

Data availability

Data will be made available on request.

References

Abnar, S., & Zuidema, W. H. (2020). Quantifying attention flow in transformers. In *Pro. ACL* (pp. 4190–4197).

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. In O. D. Suarez (Ed.), *PLoS One*, *10*(7), Article e0130140.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., et al. (2020). Language models are few-shot learners. In *Proc. NeurIPS*: *vol. 33*, (pp. 1877–1901).

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-end object detection with transformers. In *Pro. ECCV* (pp. 213–229).

Chefer, H., Gur, S., & Wolf, L. (2021a). Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *Pro. ICCV* (pp. 387–396).

Chefer, H., Gur, S., & Wolf, L. (2021b). Transformer interpretability beyond attention visualization. In *Pro. CVPR* (pp. 782–791).

Chen, L., You, Z., Zhang, N., Xi, J., & Le, X. (2022). UTRAD: Anomaly detection and localization with U-transformer. *Neural Networks*, *147*, 53–62.

Cheng, L., Fang, P., Liang, Y., Zhang, L., Shen, C., & Wang, H. (2022). TSGB: Target-selective gradient backprop for probing CNN visual saliency. *IEEE Transactions on Image Processing*, *31*, 2529–2540.

Cheng, C., Liu, W., Fan, Z., Feng, L., & Jia, Z. (2024). A novel transformer autoencoder for multi-modal emotion recognition with incomplete data. *Neural Networks*, *172*, Article 106111.

Chu, X., Tian, Z., Wang, Y., Zhang, B., Ren, H., Wei, X., et al. (2021). Twins: Revisiting the design of spatial attention in vision transformers. In *Proc. NeurIPS* (pp. 9355–9366).

Clark, K., Khandelwal, U., Levy, O., & Manning, C. D. (2019). What does BERT look at? An analysis of BERT's attention. In *Pro. ACL workshop* (pp. 276–286).

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Pro. NAACL-HLT* (pp. 4171–4186).

DeYoung, J., Jain, S., Rajani, N. F., Lehman, E., Xiong, C., Socher, R., et al. (2020). ERASER: A benchmark to evaluate rationalized NLP models. In *Pro. ACL* (pp. 4443–4458).

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In *Pro. ICLR*.

Everingham, M., Gool, L. V., Williams, C. K. I., Winn, J. M., & Zisserman, A. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, *88*(2), 303–338.

Ghiasi, A., Kazemi, H., Borgnia, E., Reich, S., Shu, M., Goldblum, M., et al. (2022). What do vision transformers learn? A visual exploration. CoRR abs/2212.06727.

Guillaumin, M., Küttel, D., & Ferrari, V. (2014). ImageNet auto-annotation with segmentation propagation. *International Journal of Computer Vision*, *110*(3), 328–348.

Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., et al. (2023). A survey on vision transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(1), 87–110.

Hao, Y., Dong, L., Wei, F., & Xu, K. (2021). Self-attention attribution: Interpreting information interactions inside transformer. In *Pro. AAAI* (pp. 12963–12971).

Hariharan, B., Arbelaez, P., Bourdev, L. D., Maji, S., & Malik, J. (2011). Semantic contours from inverse detectors. In *Pro. ICCV* (pp. 991–998).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Pro. CVPR* (pp. 770–778).

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

Kovaleva, O., Romanov, A., Rogers, A., & Rumshisky, A. (2019). Revealing the dark secrets of BERT. In *Pro. EMNLP/IJCNLP* (pp. 4364–4373).

Leem, S., & Seo, H. (2024). Attention guided CAM: Visual explanations of vision transformer guided by self-attention. arXiv preprint arXiv:2402.04563.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., et al. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In *Pro. ICCV* (pp. 10012–10022).

Ma, J., Bai, Y., Zhong, B., Zhang, W., Yao, T., & Mei, T. (2023). Visualizing and understanding patch interactions in vision transformer. *IEEE Transactions on Neural Networks and Learning Systems*, 1–10.

Montavon, G., Lapuschkin, S., Binder, A., Samek, W., & Müller, K.-R. (2017). Explaining nonlinear classification decisions with deep Taylor decomposition. *Pattern Recognition*, *65*, 211–222.

Qiang, Y., Pan, D., Li, C., Li, X., Jang, R., & Zhu, D. (2022). AttCAT: Explaining transformers via attentive class activation tokens. In *Proc. NeurIPS* (pp. 5052–5064).

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211–252.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2020). Grad-CAM: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, *128*(2), 336–359.

Serrano, S., & Smith, N. A. (2019). Is attention interpretable? In *Pro. ACL* (pp. 2931–2951).

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *Pro. ICLR*.

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. In *Pro. ICML* (pp. 3319–3328).

Vasanthi, P., & Mohan, L. (2023). A reliable anchor regenerative-based transformer model for x-small and dense objects recognition. *Neural Networks*, *165*, 809–829.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. In *Proc. NeurIPS* (pp. 5998–6008).

Vilas, M. G., Schaumlöffel, T., & Roig, G. (2023). Analyzing vision transformers for image classification in class embedding space. In *Proc. NeurIPS*.

Voita, E., Talbot, D., Moiseev, F., Sennrich, R., & Titov, I. (2019). Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Pro. ACL* (pp. 5797–5808).

Vukadin, D., Afrić, P., Šilić, M., & Delač, G. (2024). Advancing attribution-based neural network explainability through relative absolute magnitude layer-wise relevance propagation and multi-component evaluation. *ACM Transactions on Intelligent Systems and Technology*.

Xie, W., Li, X., Cao, C. C., & Zhang, N. L. (2023). ViT-CX: Causal explanation of vision transformers. In *Pro. IJCAI* (pp. 1569–1577).

Xu, L., Ouyang, W., Bennamoun, M., Boussaïd, F., & Xu, D. (2022). Multi-class token transformer for weakly supervised semantic segmentation. In *Pro. CVPR* (pp. 4300–4309).

Xu, L., Yan, X., Ding, W., & Liu, Z. (2022). Attribution rollout: a new way to interpret visual transformer. *Journal of Ambient Intelligence and Humanized Computing*, 1–11.

Yuan, L., Hou, Q., Jiang, Z., Feng, J., & Yan, S. (2023). VOLO: Vision outlooker for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(5), 6575–6586.

Yuan, T., Li, X., Xiong, H., Cao, H., & Dou, D. (2021). Explaining information flow inside vision transformers using Markov chain. In *Proc. NeurIPS workshop*.

Zaidan, O., & Eisner, J. (2008). Modeling annotators: A generative approach to learning from annotator rationales. In *Pro. EMNLP/IJCNLP* (pp. 31–40).

Zhang, N., Yu, L., Zhang, D., Wu, W., Tian, S., Kang, X., et al. (2024). CT-Net: Asymmetric compound branch transformer for medical image segmentation. *Neural Networks*, *170*, 298–311.