computers

MDPI

Article

The Explainability of Transformers: Current Status and Directions

Paolo Fantozzi [1,†] **and Maurizio Naldi** [1,2,*,†]

[1] Department of Law, Economics, Politics, and Modern Languages, LUMSA University, 00192 Rome, Italy; p.fantozzi1@lumsa.it
[2] Department of Civil, Computer Science and Aeronautical Technologies Engineering, Roma Tre University, 00146 Rome, Italy
* Correspondence: m.naldi@lumsa.it
† These authors contributed equally to this work.

Abstract: An increasing demand for model explainability has accompanied the widespread adoption of transformers in various fields of applications. In this paper, we conduct a survey of the existing literature on the explainability of transformers. We provide a taxonomy of methods based on the combination of transformer components that are leveraged to arrive at the explanation. For each method, we describe its mechanism and survey its applications. We find out that attention-based methods, both alone and in conjunction with activation-based and gradient-based methods, are the most employed ones. A growing attention is also devoted to the deployment of visualization techniques to help the explanation process.

Keywords: explainability; transformers; visual transformers; natural language processing; interpretability; deep learning

Citation: Fantozzi, P.; Naldi, M. The Explainability of Transformers: Current Status and Directions. *Computers* **2024**, *13*, 92. https://doi.org/10.3390/computers13040092

Academic Editor: Kartik B. Ariyur

Received: 8 March 2024
Revised: 28 March 2024
Accepted: 1 April 2024
Published: 4 April 2024

1. Introduction

Transformers are deep neural networks that exploit the self-attention mechanism to capture relationships between different portions of a text and have rapidly attracted interest in machine learning. Their applications span several domains [1], including natural language processing, Computer Vision [2], Audio and Speech signals [3], and Signal Processing [4]. Their popularity is making transformers become one of the most used deep learning architectures.

At the same time, the inherently opaque behavior of deep learning techniques has posed the problem of their explainability. Explainability (and other close terms such as interpretability [5,6]) concerns the attempt to overcome the inherent opacity of black-box models such as deep learning networks, but also ensemble methods such as those based on bagging (random forests) or boosting (e.g., AdaBoost or XGBoost). In general terms, we can define explainability as the capability to offer a detailed understanding of a machine learning (ML) model and its outputs. Such a need is present for all ML tasks, i.e., supervised tasks [7] such as classification, unsupervised tasks [8], and reinforcement learning [9]. The term XAI (Explainable Artificial Intelligence) has now come to describe this whole area of work. According to [10], that term was coined by Lent et al. [11] in 2004.

The need for explainability is strong for several reasons. This is not necessarily the case for all ML applications, as noted in [10]. Low-risk systems (where the consequences on human beings are minor) and well-studied high-trust systems (where we may dispense with explanations) are two examples where explainability is not an issue. Instead, Rudin herself focused on high-stakes decisions, where the use of ML models may have significant consequences on human lives, and mentioned healthcare and justice as two major fields where significant efforts have to be spent to achieve a transparent look at ML models' outputs. In some cases, there are also legal requirements to enforce explainability [12].

Though the topic of the explainability of more established deep learning architecture has been widely addressed, the relative youth of transformer architecture has generated

a more recent stream of studies on the explainability of transformers, whose number has, however, rapidly grown in parallel with the diffusion of the architecture itself. The establishment of a significant body of knowledge about the explainability of transformers suggests the need to organize it so as to identify the most relevant approaches, facilitate the approach of newcomers to the field, and highlight research gaps. Unfortunately, the few literature surveys (reviewed in Section 3) pre-date the explosion of interest for the topic and so do not account for the vast majority of the literature now existing.

We aim to fill that gap by proposing a survey of the current literature on the explainability of transformers by adopting a systematic approach based on a major scientific literature database (Scopus) and covering the whole production since the inception of the research efforts on this topic.

Our contributions are the following:

- We provide a taxonomy of explainability methods for transformers that relies on the transformer component that the method leverages;
- We describe each method and survey their applications;
- We identify the most favored methods;
- We identify the research gaps and provide some directions for future research.

This paper is organized as follows: We first describe the architecture of transformers in Section 2. Our literature review in Section 3 focuses on previous literature surveys on the same topic as ours. The literature proposing or applying explainability techniques for transformers is examined in Section 5, based on the dataset whose composition is reported in Section 4.

2. Transformers

Before dealing with the literature on the explainability of transformers in the next sections, we provide some background information about the architecture of transformers and their fields of application in this section.

The transformer architecture, introduced in [13], is a deep learning architecture, originally intended for sequences of text. Its structure is shown in Figure 1. It is composed of two different sections: an encoder and a decoder. The encoder part is responsible for compressing all the information derived from the input into a vector that is used by the decoder to build the next elements (e.g., the next pieces of text) in the sequence. The input to both sections is the embedding representation of each element of the sequence, combined with some form of positional encoding to represent the position of the element with respect to the whole sequence. The typical length of the embedding vector is 512. In most implementations, just as in the original paper, the positional encoding is carried out through a set of sine and cosine functions, whose frequency is modulated by the progressive index of the embedding vector element [14]. Each element of the embedding vector is added to the positional encoding vector. The specific procedure used to carry out embedding is not critical as long as it is the same for all the elements of the sequence.

Both the encoder and the decoder are composed of stacked layers of the same kind: a multi-head attention layer followed by some feed-forward linear layer. In the decoder modules, there is one more multi-head attention layer between the other two, which is used to map the input vector over all the sequences that make the output. There are also residual connections around each block, used to stabilize training. Each layer is the input to the next one.

Each multi-head attention block has to learn an attention-aware vector for each element of the sequence, based on the self-attention mechanism proposed in [15], which is in turn derived from the attention mechanism introduced in [16]. The attention-aware version of a vector in the sequence can be seen as the combination between the original vector and a weighted sum over all the vectors in the sequence, where the weights of the vectors are learned during training. Furthermore, the elements of the sequence are not the same in all the layers, but each of them learns a representation of them, also based on the representation

used in the previous layer. The result of this stacked computation is a non-linear function of all the elements of the sequence combined.

Formally, we define the input of each head of each layer l of a transformer architecture as a sequence h^{l-1} of vectors such that $h_i^{l-1} \in \mathbb{R}^d$ is the i-th vector in input, and we define the output as a sequence h^l of the same dimensions. Each vector h^l is computed by using a softmax function:

$$A_{i,j}^l = \text{softmax}\left(\frac{Q\left(h_i^{l-1}\right) \cdot K\left(h_i^{l-1}\right)^T}{\sqrt{d}}\right) \in \mathbb{R} \tag{1}$$

$$h_i^l = W_O^l \cdot \left(\sum_j A_{i,j}^l V\left(h_j^{l-1}\right) + h_i^{l-1}\right) \tag{2}$$

where A^l is the attention matrix of the layer l; W_O^l is the weights matrix learned by the feed-forward layer in output; and Q, K, and V are transformations defined as

$$Q(h) = W^Q \cdot h, \qquad V(h) = W^V \cdot h, \qquad K(h) = W^K \cdot h, \qquad W^Q, W^V, W^K \in \mathbb{R}^{d \times d} \tag{3}$$

with W^Q, W^V, W^K being weight matrices learned by the linear modules associated with the query, values, and keys branches, respectively. In each of the previous definitions, the bias term is dropped to simplify the notation. Furthermore, in the multi-head modules, all the head outputs are concatenated.

Since its introduction, the transformer has undergone many different experiments on variations of its attention mechanism and its architecture. One of the most influential variations of the transformer architecture is the transformer decoder presented in [17], where the encoder part of the transformer is completely dropped, leaving the decoder as the only component of the architecture. In this variation, both the input and the output are considered as part of the same sequence, with the model trained to just complete the sequence, given the first part that corresponds to the original input. Following the work conducted in [17], Radford et al. presented in [18] GPT-1, a decoder-only transformer trained using a self-supervised approach on a wide variety of texts, which was the first in a series of experiments whose last incarnation (at the current date) is GPT-4, presented by OpenAI in [19].

A different path of experiments began with the work presented in [20], where a different way was proposed to train an encoder-only transformer architecture. Indeed, they trained the model, randomly masking tokens of the input sequence and leaving it to the model to learn the correct missing tokens. Such an approach was called BERT (Bidirectional Encoder Representations from Transformers). Many different works have been derived from BERT, modifying some aspects to improve performances, such as RoBERTa, presented in [21], and DistilBERT, presented in [22].

A relevant branch of experiments focused on the choice of attention weights to be computed during training. Instead of computing the attention between all the combinations of input tokens, the attention may be computed over just a portion of the input sequence. Several proposals have been put forward, among which we can mention the following ones: *Sparse Attention* as presented in [23], *Linearized Attention* proposed in [24], and *Low-rank self-attention* suggested in [25].

Finally, though the architecture of transformers has initially been thought to process text, its application to non-text input is growing. Several experiments have considered the possibility of training multi-modal models, capable of understanding different types of input (e.g., text and images at the same time) and generating different types of output. It is worth mentioning VisualBERT described in [26], DALL-E by OpenAI, and Flamingo introduced in [27], which are all trained on both text and images. Furthermore, according to the Google Gemini Team in [28], this type of model has made it possible to achieve human expert performance on many different tasks.

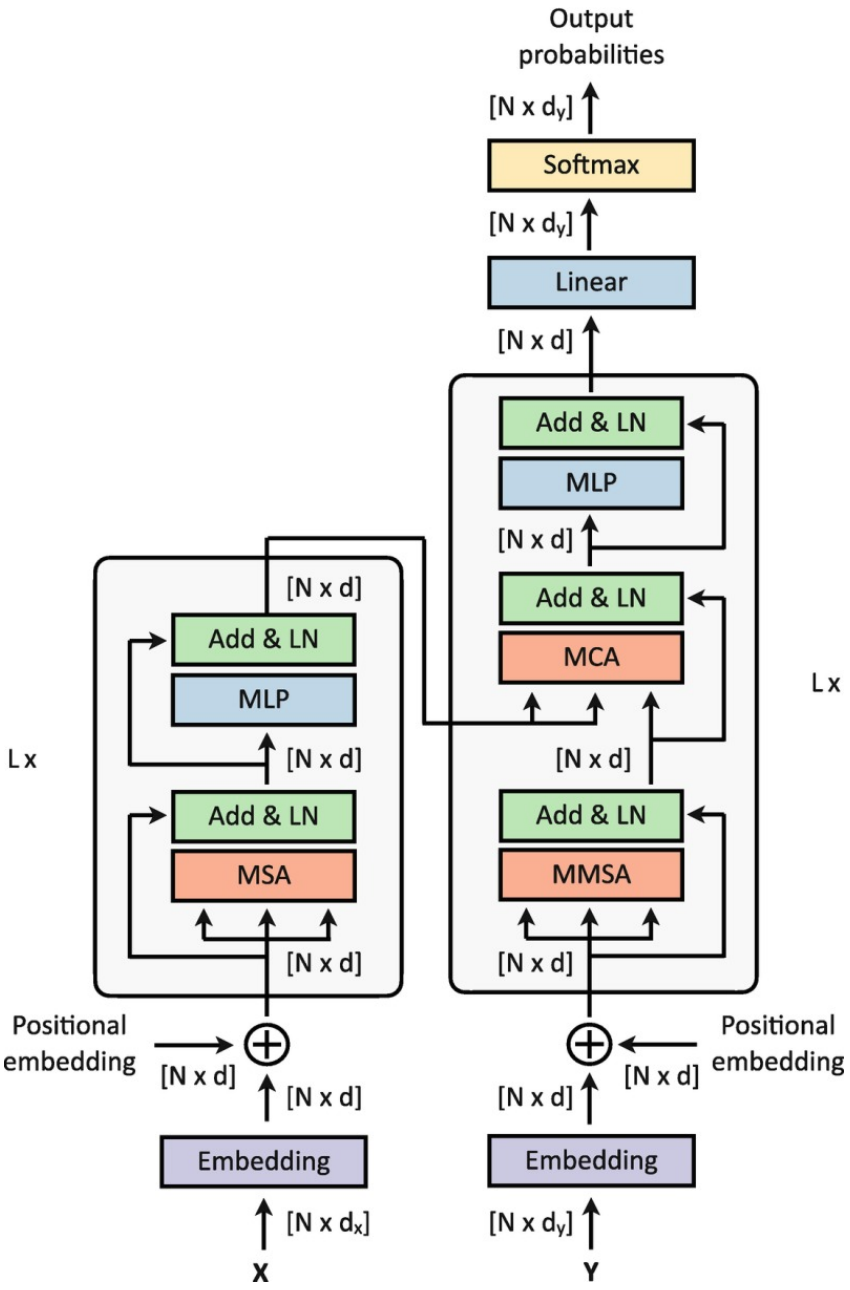

Figure 1. Transformer architecture as shown in [29] (the image is available at https://www.ncbi.nlm.nih.gov/books/NBK597474/figure/ch6.Fig3/?report=objectonly (accessed on 1 March 2024) and is licensed under the terms of the Creative Commons Attribution 4.0 International License).

3. Survey Literature

Since our aim is to provide a survey of the explainability techniques proposed for transformers, the rest of this paper is devoted to describing those techniques and reviewing the literature proposing them. In this section, we instead focus our attention on the survey literature, i.e., on those papers that have proposed a survey of explainability techniques in the past.

El-Zini and Awad claimed to provide the first survey of explainability issues in transformers [30]. Though their survey encompassed explainability for the larger class of deep learning models for natural language processing, they devoted a section to transformers. They analyzed three aspects: (1) visualization mechanisms, (2) the explainability of attention mechanisms, and (3) the explainability of BERT. As for visualization mechanisms, they analyzed papers (five in total) proposing interactive tools to help understand the

inner workings of transformers. Two of those papers dealt specifically with BERT and one with GPT-2. The explainability of attention mechanisms was explored by examining eight papers, the majority of which (six out of eight) maintained that attention mechanisms are not inherently explainable. Finally, they analyzed six papers that scrutinize BERT. The grand total was largely lower than the number of papers we found through our literature search, due to the explosion of papers in 2023, which has made many previous surveys largely incomplete.

A more restricted field of analysis was opted for by Kashefi et al., who analyzed the explainability of vision (or visual) transformers, i.e., transformers designed for image understanding rather than NLP [31]. They introduced a taxonomy of models (from which we started to propose our more comprehensive one), built around five classes:

- Common-attribution methods;
- Attention-based methods;
- Pruning-based methods;
- Inherently explainable methods;
- Non-classification tasks.

As can be seen, their taxonomy relies on a mixture of criteria, making the taxonomy non-homogeneous, since they employed the type of task carried out by the transformer alongside the component leveraged by the explainability method. The number of papers examined (falling in at least one of the previous classes) was 35.

A more limited focus was taken by Vijayakumar, who considered explainability attained through the feed-forward layer (that follows the attention layer in the transformer architecture, as shown in Figure 1) just in the context of NLP [32]. Their limited focus leads to a very small number of papers surveyed, precisely just seven.

Beyond the survey of explainability methods, Brasoveanu and Andonie proposed instead a survey of papers dealing with the visualization of transformer operations [33]. Though they did not explicitly address explainability, visualization is tightly related to explainability since visualizing the operations of the neural network helps understand the role and relevance of the different contributions to the final outcome. They made a distinction between focused and holistic visualizers, where the former center on a single component of the machine, e.g., attention, while the latter examine the behavior of the transformer as a whole. Though they claimed to have examined 50 papers, they decided to restrict their attention to papers proposing NLP applications. Their final selection was made of 12 papers on focused visualizers and 6 papers on holistic visualizers. Again, the latest flurry of papers on the explainability of transformers urges the analysis of a wider body of literature.

Vision Transformers were, again, the focus of the survey conducted in [34]. The survey focused on post hoc XAI methods and proposed a classification of the taxonomies adopted for XAI methods. The authors distinguished between methods originally thought for CNNs (Convolutional Neural Networks) and methods specifically developed for Vision Transformers. The number of papers surveyed was 9 for the former category and 14 for the latter. Though the panorama was somewhat restricted, the survey differs significantly from the others in its effort to evaluate XAI methods on a common ground, employing the ImageNet Large-Scale Visual Recognition Challenge dataset [35] and adopting several metrics to carry out the evaluation using five criteria: faithfulness, complexity, randomization, robustness, and localization.

After examining the body of literature represented by surveys, we can conclude that ours differs from those that appeared so far in one or more of the following aspects:

- Time coverage;
- Extension, as for the number of papers analyzed;
- Extension, as for the type of application analyzed.

As for time coverage, all papers stopped their coverage at (or earlier than) 2022, missing the explosion of the literature in 2023, which we instead covered in full. The

number of papers under analysis lay below 10 in [30,32] and stopped at 35 even in the most comprehensive survey [31], while we analyzed 95. The type of application is limited to NLP applications in [32] and to visual transformers in [31,33,34], while we considered all application domains.

4. Dataset

We extracted the literature of interest by querying the well-known Scopus database. The query used the terms *explainability* AND *transformers*. We obtained 279 papers (the database was accessed for the last time on 16 February 2024). We proceeded to clean that dataset using a systematic approach following the PRISMA guidelines [36,37]. The selection flowchart is shown in Figure 2. We removed first the items that represented whole conferences rather than individual papers. We then examined the abstracts to select just those papers actually dealing with explainability techniques applied to the deep learning architecture named transformer. Finally, after examining the full text of those papers, we removed another group of papers for two major reasons: they either did not deal with transformers, though they mention the architecture in the abstract, or they did not propose or analyze XAI techniques.

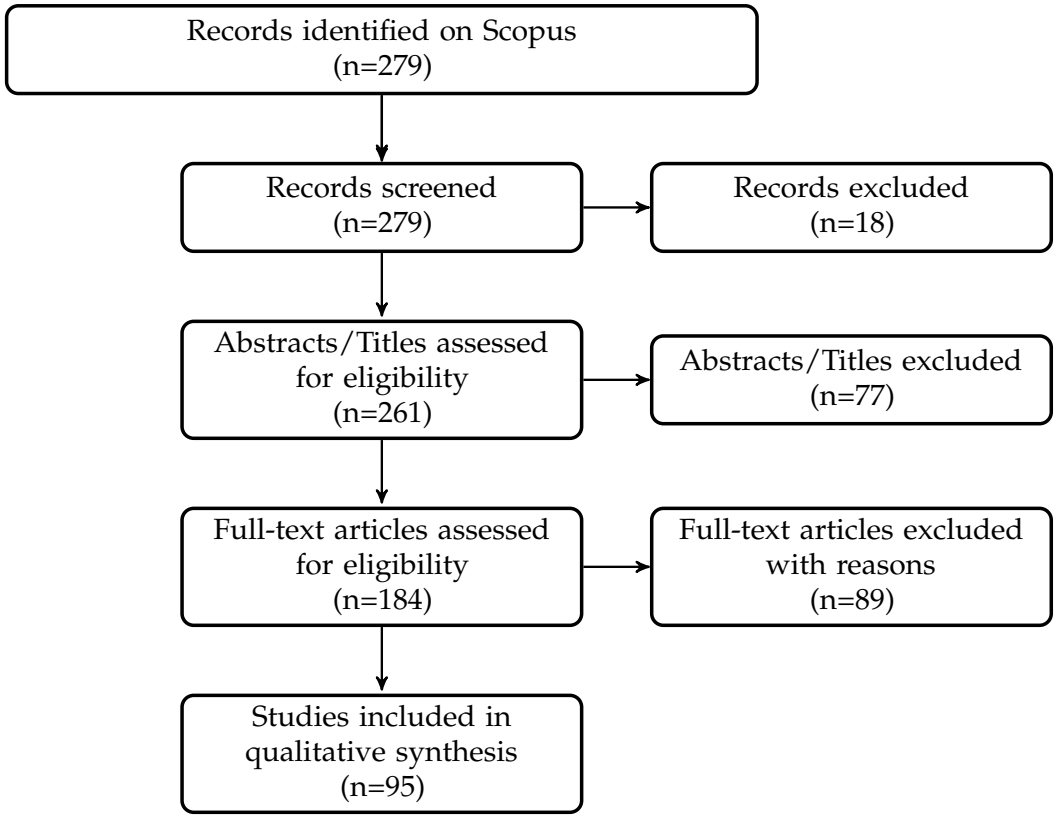

Figure 2. Systematic selection flowchart.

We can first examine the evolution of the papers satisfying the search criteria over time in Figure 3. The figure for 2024 up to the last time of access to Scopus is projected to the end of the year to make it comparable with the other years. The explosion of papers in the last two years is well visible and is a major reason to have an up-to-date survey of the subject.

Not all the papers introduced methodological innovations. Some papers introducing new XAI methods did so with no specific reference to transformers, though their methods could be applied to transformers. On the other hand, the vast majority of papers dealing with XAI for transformers analyzed the application of existing techniques, maybe with minor variants. We reported the innovative papers in Figure 4. Again, we projected the

data for 2024, though the statistical reliability of such a projection is quite low since just a few innovative papers have appeared in 2024 so far. Though not explosive, the increase in 2023 is anyway visible here as well. The interest in the subject is growing.

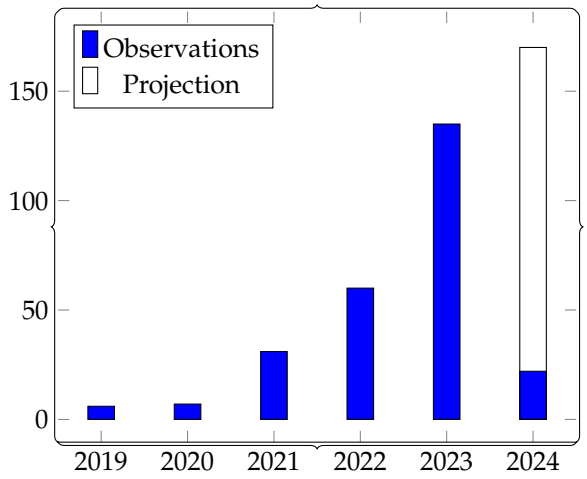

Figure 3. Number of papers over time.

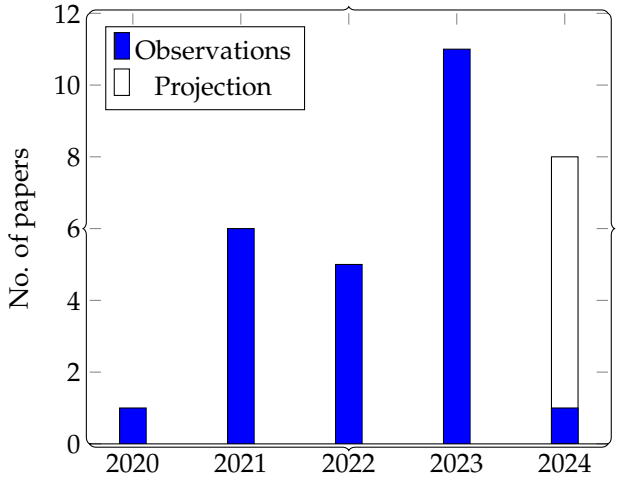

Figure 4. Number of papers introducing new methods by year of publication.

The distribution of papers proposing innovative methods by country in Figure 5 shows some leadership by China and the U.S.A., but with European countries (Italy, Germany, and the UK) as good runner-ups.

If we consider the domain of application by type of input data in Figure 6, we can surely observe the large predominance of textual data and images. Though the leading role of the former type is well expected, since transformers have been initially proposed for NLP applications, image applications are now on par, marking the recent extension of transformers to visual data.

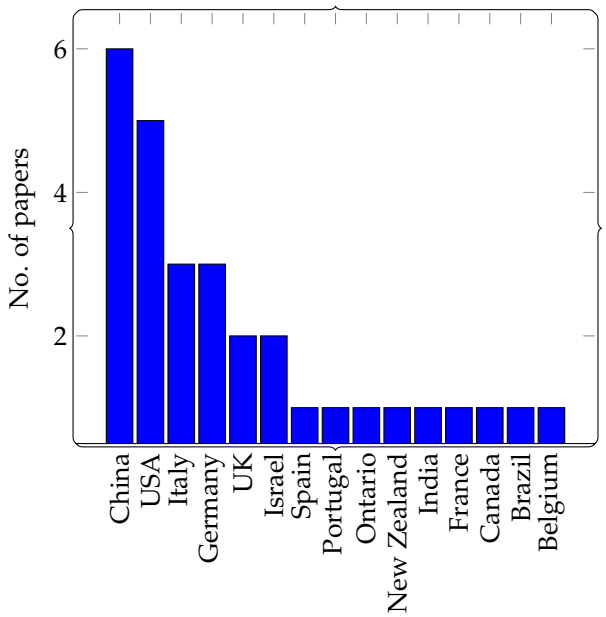

Figure 5. Number of papers introducing new methods by country of authors.

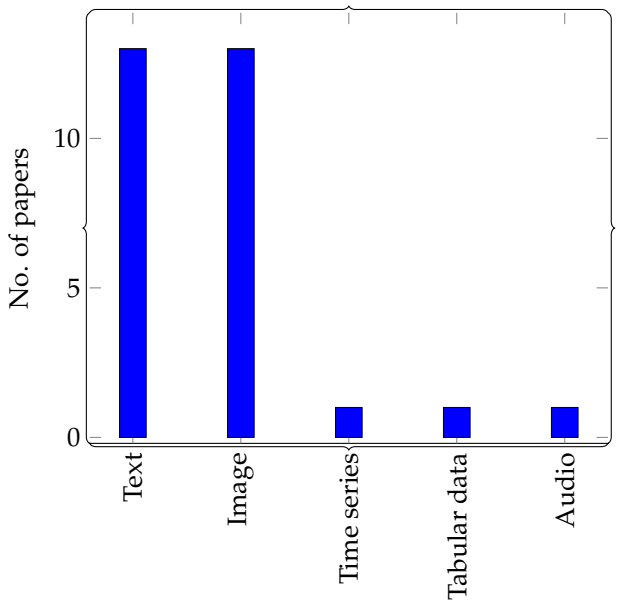

Figure 6. Number of papers introducing new methods by type of input.

5. Methodologies for Explainability

In Section 3, we described the survey literature. That body of literature proposed a panorama of the existing explainability techniques. In this section, we go on to describe the literature proposing new explainability techniques for transformers. In doing so, we propose a classification of those techniques according to the architecture components they employ to explain the transformer's output. Our classification modifies that introduced in [31], which included mixed criteria. Our classification employs the following classes and their combinations:

- Activation;
- Attention;
- Gradient;
- Perturbation.

Hereafter, for each class, we include the original formulation of the technique, though it was not expressly devised for transformers but later applied to them with some modifications, and describe the main papers that have adopted it.

5.1. Activation-Based Methods

One of the main ideas to identify the contribution of each input feature to the output is to use the activation of the neurons, going through the network back to the input. Developing this idea, Bach et al. presented in [38] the Layer-wise Relevance Propagation (LRP). This technique is based on the assumption that the activation of a neuron signals its relevance for the output. Also, the relationship between different layers is represented by the relevance of a neuron in a layer being a linear combination of the relevances of the neurons in the previous layer. This relationship allows us to track neuron relevances from the output back to the input. Furthermore, the basic method assumes that the network uses ReLU activations in all layers.

Using a_i as the dummy variable describing the activation of neuron i, $w_{i,j}$ as the weight in the layer from the input i to the output j, and R_i^l as the relevance of the neuron i in the layer l, we consider the following relation to hold, which allows us to compute the relevance of any neuron in a layer based on the relevances of all neurons in the subsequent layer:

$$R_i^{l-1} = \sum_j \frac{a_i \cdot w_{i,j}}{\sum_k a_k \cdot w_{k,j}} \cdot R_j^l \tag{4}$$

We can apply this backtracking relationship starting from the output layer and going through the network back to the input layer, hence identifying the relevance of each input feature.

Even though the LRP method was developed to be applied to CNNs for images (which explains the assumption of ReLU activation), it has been explored and modified to be applied to other types of NNs. Ding et al. applied LRP for machine translation tasks in attention-based encoder–decoder RNNs [39]. Unlike the original method, they computed the relevance score for input vectors instead of single features. So, the relevance score for an embedding vector corresponds to the sum of the relevance scores of its features. Furthermore, they proposed to ignore the non-linear activation functions during LRP computations, based on the assumption that the choice of non-linear functions is irrelevant for LRP [38].

The application of LRP to transformers requires three more issues to cope with: different activation functions, skip connections, and matrix multiplications. Indeed, comparing a CNN architecture with that defined in Equations (1)–(3) allowed us to spot many differences; first of all, the multiplication $Q \cdot V$ and the term $\cdots + h_i^{l-1}$ used to implement the skip connections. Following the implementation of [39], Voita et al. presented Partial-LRP to identify the most relevant heads in a transformer-based model and prune the least relevant ones [40]. In applying LRP to their models, they considered a value $R_{u \leftarrow v}$ as the relevance of neuron u for neuron v, which can be defined as

$$R_{u \leftarrow v} = \sum_{z \in \mathrm{OUT}(u)} w_{u \to z} \cdot R_{z \leftarrow v} \tag{5}$$

where $\mathrm{OUT}(u)$ is the set of nodes directly connected to u in the next layer, and $w_{u \to v}$ is the weight ratio that measures the contribution of u to v, defined as

$$w_{u \to v} = \begin{cases} \dfrac{W_{u,v} \cdot u}{\sum_{u' \in \mathrm{IN}(v)} W_{u',v} \cdot u'} & \text{if } v = \sum_{u' \in \mathrm{IN}(v)} W_{u',v} \cdot u' \\ \dfrac{u}{\sum_{u' \in \mathrm{IN}(v)} u'} & \text{if } v = \prod_{u' \in \mathrm{IN}(v)} u' \end{cases} \tag{6}$$

where $W_{u,v}$ is the weight learned between neurons u and v, and $\mathrm{IN}(u)$ is the set of nodes directly connected to u in the previous layer.

Chefer et al. in [41] proposed a variation of LRP where the issue of different activation functions is settled just considering the $a_i \cdot w_{i,j}$ elements that turn out to be non-negative. Their analysis of the two other issues shows that both operations could be seen as Relevance Propagation through two different tensors instead of a single one. So, they expanded the definition of relevance for both tensors u and v as

$$\begin{aligned} R_i^{u^{l-1}} &= \sum_j \frac{u_i \cdot v_{i,j}}{\sum_k u_k \cdot v_{k,j}} \cdot R_j^l \\ R_i^{v^{l-1}} &= \sum_j \frac{v_i \cdot u_{i,j}}{\sum_k v_k \cdot u_{k,j}} \cdot R_j^l \end{aligned} \tag{7}$$

Finally, they introduced normalization:

$$\begin{aligned} \overline{R}_j^{u^{l-1}} &= R_j^{u^{l-1}} \cdot \frac{\left|\sum_j R_j^{u^{l-1}}\right|}{\left|\sum_j R_j^{u^{l-1}}\right| + \left|\sum_j R_k^{v^{l-1}}\right|} \cdot \frac{\sum_i R_i^l}{\sum_j R_j^{u^{l-1}}} \\ \overline{R}_k^{v^{l-1}} &= R_k^{v^{l-1}} \cdot \frac{\left|\sum_j R_k^{v^{l-1}}\right|}{\left|\sum_j R_j^{u^{l-1}}\right| + \left|\sum_j R_k^{v^{l-1}}\right|} \cdot \frac{\sum_i R_i^l}{\sum_j R_k^{v^{l-1}}} \end{aligned} \tag{8}$$

Even though this relevance score could be used to provide explanations for each attention layer (as in standard LRP), they used it as a construction block for a different score that computes both LRP and Attention Rollout, as described in more detail in Section 5.5.1.

In order to improve the distinction between the positive and negative relevance of neurons, Nam et al. in [42] proposed a variation of LRP called Relative Attributing Propagation (RAP), which uses normalization before the propagation of positive and negative relevances in each layer separately.

Since the results of LRP are class-independent, many methods have tried to differentiate the results of LRP to represent the classes. Gu et al. in [43] presented Contrastive Layer-wise Relevance Propagation (CLRP), which is based on the idea of computing LRP both for the class of interest and for the aggregation of the other classes, keeping only the positive differences between the two classes of relevances obtained.

In image classification, one of the most interesting aspects of prediction is understanding which image features were most influential in predicting the class. To estimate this type of influence, Zhou et al. introduced the Class Activation Mapping (CAM) technique in [44]. CAM considers a CNN with a global average pool just before the last fully connected layer (which could employ, e.g., a softmax function for classification and a ReLU activation function for regression). Its application is quite limited, given the large number of architectures other than CNN that have been proposed in the literature. The method consists of estimating the score for each area of the input image by multiplying the activation of each filter of the last convolutional layer (before the global average pool) and the weight learned for the average of the filter with respect to the neuron representing the class in the output layer. That multiplication outputs a matrix with the same size as the last convolutional layer for each filter. Summing them provides us with a relevance matrix for a single output class. The last step consists of visualizing this relevance matrix over the original input, resizing (usually with some form of interpolation) the relevance matrix to the same size as the original input image. Formally, given the last convolutional layer of a CNN architecture with the activation matrix $V^{n \times m \times F}$, composed of F filters of size $n \times m$, followed by a global average pool and a vector representing the output classes connected by a weight matrix W, we compute the relevance score $R_{x,y,c}$ for a super pixel at position (x, y) for the class c as

$$R_{x,y,c} = \sum_f \left(W_{f,c} \cdot V_{x,y,f} \right) \tag{9}$$

A super pixel is a segment of the matrix after reshaping by applying the filters of the architecture, basically corresponding to a patch of the input image.

A different idea was proposed by Ferrando et al. in [45], with a method called ALTI (Aggregation of Layer-wise Token-to-Token Interactions). The method computes the contribution of each component of a transformer block to the output of the block. Roughly speaking, the idea is to compute the difference between the component $A^l_{i,j} V(h^{l-1}_j)$ from Equation (2) and h^l_i as the contribution of the j-th component to the i-th output. All the matrices composed using these differences are then combined using the same rules of Attention Rollout (explained in Section 5.2).

Focusing on the differences appearing in the output due to different inputs, Li et al. in [46] proposed a method (subsequently) called Input Erasure that masks part of the input with 0s to measure the contribution (the relevance) of this part to the same output (it could be even just a single embedding dimension).

A simple approach was proposed by Kim et al. in [47] with a method called Concept Activation Vectors (CAVs). They considered the activations of a layer for many input samples, both for the target class and the non-target class (in a binary problem). Afterward, they trained a linear model on the activations to distinguish between the target class and the non-target class. The linear model weights were the relevance scores for the class with respect to the features.

Muhammad and Yeasin in [48] proposed a method called Eigen-CAM, inspired by CAM, that is based on Single Value Decomposition. They combined the weight matrices from the first k convolutional layers, multiplying them by the input image matrix. The saliency map consisted of the projection of the matrix just computed on the first eigenvector.

Following the techniques described so far, many papers applied them in different contexts, often employing several techniques at the same time to look for the best one.

For example, Mishra et al. in [49] compared different methods (LRP and perturbation methods like LIME and SHAP, described in Section 5.4) to explain models for hate speech detection. Instead of straightforwardly evaluating models, Thorn Jakobsen et al. in [50] turned their attention to the datasets employed to evaluate explanability methods, proposing new datasets and using LRP and Attention Rollout (see Section 5.2). Other authors focused on LRP for several purposes. Yu and Xiang in [51] proposed a model for neural network pruning, visualising relevances using LRP. Chan et al. proposed a new method to perform early crop classification by exploiting LRP in [52].

CAM was instead the method chosen in [53] to explain remote sensing classification performed through a transformer-based architecture.

The ALTI method was instead chosen and slightly modified in [54] by Fernando et al., who proposed a variant called ALTI-logit, where each component $A^l_{i,j} V(h^{l-1}_j)$ is multiplied by the matrix just before the final softmax function in the network. They further proposed Contrastive ALTI-logit, where the difference between two different tokens is measured at the output, subtracting the results of ALTI-logit for the two tokens. They also compared their methods with Contrastive Gradient Norm and Contrastive InputXGradient.

CAVs were instead applied by Madsen et al. in [55] on an EEG classification model.

A comparison for (multi-modal) transformers was carried out in [56], namely between Optimal Transport, which considers activations of different input types, and Label Attribution, which is a variation of TMME (see Section 5.5.4). An even wider comparison was carried out by Hroub et al. in [57], where different models are employed for pneumonia and COVID-19 existence prediction from X-rays. The set of methods included Grad-CAM, Grad-CAM++, Eigen-Grad-CAM, and AblationCAM to produce saliency maps.

Another group of papers focused on visualization techniques. Alammar in [58] presented a tool (Ecco) to provide different visualization techniques for transformers. Each of them could be classified into one of the two classes: Gradient $\times$ Input or a function of

activation (in this case, dimensionality reduction over responses is also performed). Van Aken et al. in [59] presented VisBERT, a visualization tool for BERT-like architectures, which is an activation-based method (they used the responses of each layer) followed by dimensionality reduction techniques (t-SNE, PCA, and ICA) to project input tokens on a 2D plane.

Gao et al. in [60] proposed a new architecture for table explanation, supplying three different methods of explanation: local explanation, global explanation, and structural explanation. The local explanation consists of computing output embeddings of text and extracting many subsequences, each of them being represented as the mean of the embeddings that compose it minus the embeddings of the CLS token. Afterward, they multiplied the matrix obtained by the weight matrix of the last layer just before applying a sigmoid activation. The results were the relevance scores of the subsequences. The global explanation consisted of computing the cosine similarity between the embeddings of the CLS (cross-lingual summarization) token of each sample in the dataset and assigning a relevance score to each of them accordingly. The structural explanation consisted of building a graph from the dataset, computing the embeddings of nodes and multiplying the embeddings by each other (CLS embeddings multiplied by its neighbors), and then (after multiplying the normalized version again for the neighbors' embeddings) aggregating (by summing them) the scores of each neighbor and using the results as relevance scores.

5.2. Attention-Based Methods

Since the introduction of the attention mechanism in [16], attention weights have been one of the go-to indicators to estimate explanations. In [61], Abnar et al. introduced *Attention Rollout* and *Attention Flow* techniques. They share the same ground assumptions but differ in the information flow mechanism through the neural network. The assumption they share is that the attention in the last layer cannot be considered a proxy for explanation. Considering instead the attention in the first layers, we can use it to measure the contribution of each token to the result. They also underlined the importance of the residual connection with respect to the information flow, so instead of using just attention weights, they augmented the attention weights matrix with a layer l as

$$A^l = 0.5 \cdot W^l_{att} + 0.5 \cdot I \tag{10}$$

where W^l_{att} is the average of all the attention weight matrices in all the heads of the layer l of the transformer.

In the *Attention Rollout* technique, a chain of cumulative attention matrices is formed by multiplying the attention matrix A^l of the l-th layer by the attention matrices of the subsequent layers:

$$\tilde{A}^l = \begin{cases} A^l & \text{if it is the last layer} \\ A^l \tilde{A}^{l-1} & \text{otherwise} \end{cases} \tag{11}$$

The result of the multiplication can be easily shown as a heatmap of the relevance of each token in the input with respect to each token in the output.

In the *Attention Flow* technique, the neural network is seen as a graph whose edges are weighed by the attention weights A^l of the pertaining layer. Considering the weights as capacities, the input tokens as source nodes (one at a time), and the output tokens as sink nodes (again one at a time), we can compute the max flow of the network and consider it as the relevance score for the pair (source and sink).

When we come to the application of attention-based techniques, we can recognize three streams: (1) the papers proposing the usage of attention weights; (2) the papers using the Attention Rollout that we described above; and (3) the papers exploiting visualization techniques for attention weights.

In the first group, we find Renz et al., who proposed two different models for route planning in complex environments using the sum of attention weights as relevance scores [62]. Feng et al. proposed a model for early stroke mortality prediction, using

attention weights as relevance scores [63]. A more complex function of attention weights was employed by Trisedya et al. to explain the output in knowledge graph alignment [64]. Applications in the medical field were considered by Graca et al., who proposed a framework for Single Nucleotide Polymorphisms (SNPs) classification, using attention weights to explain the classification [65]; by Kim et al., who used attention weights to score text in input and explain decisions taken by a model dedicated to medical codes prediction [66]; and by Clauwaert et al., who focused on automatic genomics transcription, analyzing the attention weights of the trained model to prove the specialization of each head with respect to some input feature [67]. Sebbaq and El Faddouli proposed a new architecture to perform a taxonomy-based classification of e-learning materials, using attention weights for explainability [68]. In their model for sequential recommendation, Chen et al. also relied on attention weights for explainability [69]. An aggregation of attention weights was employed by Wantiez et al. to explain the results of their architecture for visual question answering in autonomous driving [70]. The context considered by Ou et al. to use attention weights was instead next-action prediction in reinforcement learning [71]. An aggregation over all attention weights of all heads in all layers was employed by Schwenke et al. to process time series data using symbolic abstraction [72,73]. Finally, Bacco et al. trained a transformer to perform sentiment analysis, using a function of the attention weights to select the input sentences in input that better justify the classification [74]. The same subject was more extensively dealt with in [75]. Humphreys et al. proposed an architecture for predicting defects, using the sum of attention weights over all layers [76]. Attention weights were employed for searching in a transformer-based model dedicated to multi-document summarization in [77].

The following six papers used the Attention Rollout technique. Di Nardo et al. proposed a transformer-based architecture for visual object-tracking tasks, using Attention Rollout for explainability [78]. Cremer et al. tested an architecture on 3 datasets for drug toxicity classification [79]. A variation of Attention Rollout was employed by Pasquadibisceglie et al. to generate heatmaps in a framework for next-activity prediction in process monitoring [80]. Attention Rollout was employed in conjunction with Grad-CAM (see Section 5.3) by Neto et al. to detect metaplasia in upper gastrointestinal endoscopy [81]. Both Attention Rollout and LRP were tested by Thorn Jakobsen et al. on new datasets in [50]. Finally, Komorowski et al. compared LIME, Attention Rollout, and LRP-Rollout (see Section 5.5.1) for a model dedicated to COVID-19 detection from X-ray images [82].

A large group of papers have addressed the use of the visualization of attention weights to help explain the outcome of transformers. Fiok et al. used both BertViz (a visualization tool for attention weights) and TreeSHAP (a variation of SHAP for tree-based models) [83]. Again, Tagarelli et al. employed BertViz after training a BERT-like model on the Italian Civil Code [84]. Lal et al. proposed a tool to explain transformers' decisions by visualizing attention weights in many ways, including dimensionality reduction [85]. Dai et al. adopted the visualization of the attention weights to explain a classification model to infer personality traits based on the HEXACO model [86]. Gaiger et al. considered a general transformer-based model [87]. Zeng et al. used the visualization of attention weights to explain a new framework for DNA methylation sites prediction [88]. Textual dialogue interaction was instead the application of interest in [89]. Ye et al. employed attention weights visualization to classify eye diseases from medical records [90]. Neuroscience was the domain of application considered in [91], where a new architecture for brain function analysis was proposed, and [92], where a new architecture was proposed based on the graph representation of neurons from fMRI images to predict cognitive features of the brain. Sonth et al. trained a model for driver distraction identification [93]. Kohama et al. proposed a new architecture for learning action recommendations in [94]. Wang et al. proposed a new architecture for medical image segmentation, using visualizations of both attention weights and gradient values to explain the output [95]. Kim et al. proposed an architecture for water temperature prediction [96]. Monteiro et al. proposed an architecture for a 1D binding pocket and the binding affinity of drug–target interaction pairs prediction

in [97]. Finally, Yadav et al. compared different explanability methods (LIME, SHAP, and Attention visualization) for hate speech detection models in [98].

A related research stream considered the use of transformers for images, i.e., visual transformers. Ma et al. computed an indiscriminative score for each patch of an image as a function of the attention weights in all the layers [99].

5.3. Gradient-Based Methods

Most of the training algorithms for neural networks are based on some form of gradient backpropagation from the loss function to the input. Many explainability methods are based on different functions of the gradient computed at different points in the neural network.

One of the first works using this approach was by Simonyan et al. in [100]. They presented a method (subsequently) called saliency, which computes the gradient of Y^c with respect to the input x. Employing the gradient in the linear approximation of Y^c in a neighborhood of x via its Taylor series is analogous to interpreting the coefficients in a linear regression model as a measure of feature importance. Furthermore, another work presented by Springenberg et al. in [101] introduced a class of methods that included Guided Backpropagation. This method consists of a forward pass through a CNN to reach a selected layer and then, after zeroing all the features but one, a backward pass to the input (filtering out all the non-positive pixels) to compute the relevance of the feature. After those papers, Kindermans et al. in [102] proposed to scale the scores obtained with saliency by multiplying them by the input in a method (subsequently) called InputXGradient.

Extending the work in [100], Yin and Neubig in [103] computed the gradient of the difference for two different outputs with the same input (formally $\frac{\partial(Y^c - Y^{c'})}{\partial x}$) as the saliency score, and they called it the Contrastive Gradient Norm. They also used the same gradient to extend the work in [102], calling it the Contrastive InputXGradient.

To generalize the CAM technique, Selvaraju et al. introduced in [104] the Gradient-weighted Class Activation Mapping (Grad-CAM) technique. The most important advantage of Grad-CAM with respect to CAM is the compatibility of the method with any CNN-based architecture. The method consists of backpropagating the output until the last convolutional layer. The gradients we propagated back are averaged to obtain a vector with a size equal to the number of filters. So, we use this vector just like the learned weights in the CAM method, multiplying them by the activations of the filters in the last convolutional layer. As the last step, we apply ReLU over the results of the linear combinations to filter out negative scores. Formally, given the last convolutional layer of a CNN architecture with the activation matrix $V^{n \times m \times F}$, composed by F filters of size $n \times m$, and the final output for a class c represented as Y^c, we obtain:

$$R_{x,y,c} = \text{ReLU}\left(\sum_f \left[\left(\frac{1}{F}\sum_{i,j}\frac{\partial Y^c}{\partial V_{i,j,f}}\right) \cdot V_{x,y,f}\right]\right) \tag{12}$$

Grad-CAM++ presented in [105] is a variation of Grad-CAM. Unlike GRAD-CAM, the ReLU is moved to the partial derivative, and different coefficients are used for each combination of position, filter, and class. These coefficients are computed as a function of the gradients backpropagated from the last layer of the network (before the activation function).

We can now see the papers employing one or more of the techniques described so far. Grad-CAM alone was employed by Sobahi et al., who proposed a model to detect COVID-19 by using cough sound recordings [106]; Thon et al., who proposed a model to perform a 3-classes severity classification of COVID-19 from chest radiographs [107]; and Vaid et al., who trained a model for ECG 2D representation classification [108]. Wang et al. proposed a transformer-based architecture for medical 3D image segmentation, using Grad-CAM++ in [109].

More often we find Grad-CAM employed with other techniques in a comparative fashion. Wollek et al. compared TMME and Grad-CAM for pneumothorax classification from

chest radiographs [110]. Neto et al. employed Grad-CAM and Attention Rollout to explain a model for metaplasia detection in upper gastrointestinal endoscopy [81]. Thakur et al. compared LIME and Grad-CAM for plant disease identification from leaves images [111]. Kadir et al. compared Soundness Saliency and Grad-CAM for image classification [112]. Vareille et al. employed a host of methods (SHAP, Grad-CAM, Integrated Gradients, and Occlusion) for multivariate time series analysis [113]. Hroub et al. compared different models for pneumonia and COVID-19 prediction from X-rays, using Grad-CAM, Grad-CAM++, Eigen-Grad-CAM, and AblationCAM to produce saliency maps [57].

A wider selection, not including Grad-CAM, was employed in other papers. Cornia et al. proposed a method to explain transformers' decisions in visual captioning by applying three different gradient-based methods (saliency, Guided Backpropagation, and Integrated Gradients) [114]. Poulton et al. applied saliency, InputXGradient, Integrated Gradient, Occlusion, and GradientSHAP to explain the decisions of transformers concerning the automatic short-answer grading task [115].

Finally, visualization techniques were considered by Alammar, who presented a tool (Ecco) to provide different visualization techniques for transformers [58]. Wang et al. proposed a new architecture for medical image segmentation, using the visualization of both attention weights and gradient values to explain the output [95].

As an example of the results that can be obtained with such methods, a visualization dedicated to image classification can be seen in Figure 7. In this image, each pixel is colored according to the score assigned by the different methods (most of them already explained in this section) so as to compare them. This type of visualization, consisting just of a heatmap over the input image, is usually called a saliency map.

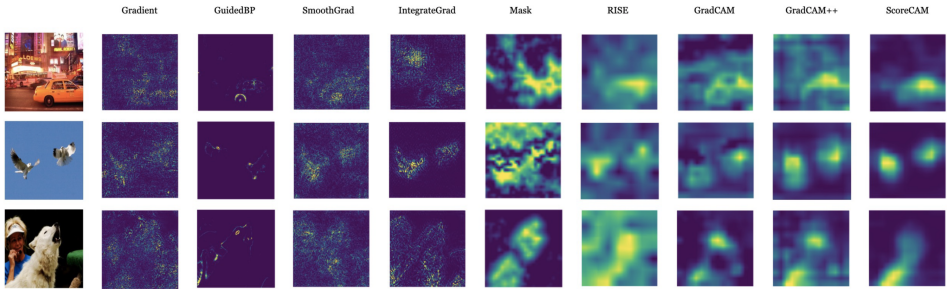

Figure 7. Visualization results of Vanilla Backpropagation, Guided Backpropagation, SmoothGrad, IntegrateGrad, Mask, RISE, Grad-CAM, Grad-CAM++, and Score-CAM [116] (the image has been taken from the open access version of the paper available at https://openaccess.thecvf.com/content_CVPRW_2020/papers/w1/Wang_Score-CAM_Score-Weighted_Visual_Explanations_for_Convolutional_Neural_Networks_CVPRW_2020_paper.pdf (accessed on 1 March 2024)).

5.4. Perturbation-Based Methods

The perturbation approach identifies the relevance of a portion of the input by masking it and checking the consequences on the output.

Zeiler and Fergus in [117] introduced an approach (subsequently) called Occlusion, where part of the input is masked with 0s and the difference in the output is measured. So, moving the Occlusion, we can use the differences in the output as a measure of the relevance of the masked part of the input.

Comparing neural networks and linear models, it is known that even though neural networks provide better results, linear models are much more easily explainable. Thinking of this fundamental difference, Ribeiro et al. introduced Local Interpretable Model-agnostic Explanation (LIME) in [118], trying to build a linear model to explain the decision taken by the neural network by lightly perturbing the input and measuring the difference in the output. Formally, given a function of probability f with respect to a class (the model we are trying to explain), a class of explainable models G (it could be the class of linear models), a function π_x that measures the proximity distance with respect to x, and a function Ω to

measure the complexity of a model of the class G, we try to find a model M for the input x defined as

$$M(x) = \arg\min_{g \in G} \ell(f, g, \pi_x) + \Omega(g) \tag{13}$$

where the function ℓ returns a measure of how unfaithful the model g is in approximating f in the locality defined by π_x.

The definition is very general and could be arranged in many different ways. For the sake of clarity, we take the class of linear models as an example for G, defined as $g(z) = w_g \cdot z$. Considering this type of model, we would define π_x as

$$\pi_x(z) = e^{\frac{-D(x,z)^2}{\sigma^2}} \tag{14}$$

with the function D defined as a distance function (e.g., cosine distance for text and L2 distance for images) and σ as a weight factor. Furthermore, we would define the function ℓ as

$$\ell(f, g, \pi_x) = \sum_{z, z' \in \mathbb{Z}} \pi_x(z) \cdot (f(z) - g(z'))^2 \tag{15}$$

with $\mathbb{Z}$ defined as the set of samples obtained by perturbing the initial input x.

It is important to notice that this type of technique guarantees a faithful local explanation but not a global one.

Lundberg et al. introduced SHAP in [119], a method based on the game-theory-based notion of Shapley values developed by Shapley in [120]. In their work, Lundberg et al. connected pre-existing methods (such as LIME and DeepLIFT) to Shapley values. They proved that LIME could return valid Shapley values after some variations. First of all, the perturbing method should introduce random masking over the features, replacing the missing ones with values sampled from a marginal distribution computed on the original input. The masking is represented by the function h_x. The proximity distance should be defined as

$$\pi'_x(z) = \frac{M - 1}{\binom{M}{|z|} |z| (M - |z|)} \tag{16}$$

where $\mathbb{Z}'$ is the set of masked samples; $z \in \mathbb{Z}'$, $|z|$ is the number of unmasked features; and M is the max number of unmasked features among the samples in $\mathbb{Z}'$. The model to be used should be a weighted linear regression model defined as

$$g'(z) = w_0 + \sum_j w_j \cdot z_j \tag{17}$$

Finally, the loss function ℓ' is defined as

$$\ell(f, g', \pi'_x) = \sum_{z \in \mathbb{Z}'} \pi_x(z) \cdot [f(h_x(z)) - g'(z)]^2 \tag{18}$$

A different approach, called Anchors, was proposed by Ribeiro et al. in [121]. They focused on the subset of input features that leaves the same output classification (with a high probability) even after the change in the remaining features.

Petsiuk et al. in [122] presented a method called RISE (Randomized Input Sampling for Explanation), which is suitable for any image classification model. It generates many random masks that are applied to the input image. The probability of classification is measured for each masked image, and then the relevance map is composed as a weighted sum of the masks (with respect to the probabilities measured in the output) after applying a normalization with respect to how many times a pixel was in a mask.

Gupta et al. in [123] propose a method (subsequently) called Soundness Saliency, which consists of learning a matrix (with the same size of the input image) used to mask the original input, such that the expectation of the negative logarithm computed on the

classification output probability is minimum. Unlike other methods, the values used to replace the masked pixels are taken from an image randomly picked from the training set. The saliency map will correspond to the learned masking matrix.

A variation of CAM, called AblationCAM, was proposed by Desai and Ramaswamy in [124], also inspired by Grad-CAM but without using gradients. They predicted the probability of an output class for an input image, considering the last layer activation matrix, just before the classification layer. They computed the relative increase in the output probability for each pixel (of each filter) in the last layer, zeroing the corresponding activation and predicting again the output probability. After that, they computed a saliency map (reshaped with respect to the input image) composed by the activation of a pixel times the percentage computed for the same pixel.

When we examine the papers reporting the application of perturbation-based methods, we see that the great majority are papers using the two best-known methods: LIME and SHAP. In most cases, they use either alone.

As for LIME, Mehta et al. applied LIME to explain the decisions of BERT-like models in a hate speech detection task [125]. Rodrigues et al. extended LIME to meta-embedding input in a model for a semantical textual similarity task with a meta-embedding approach [126]. Janssens et al. employed LIME when comparing different models to detect rumors in tweets [127]. Chen et al. compared different models to perform Patient Safety Events classification using a proprietary dataset [128]. Collini et al. proposed a framework for online reputation and tourist attraction estimation, explaining the output with LIME [129]. Finally, Silva and Frommholz employed LIME as a model to perform multi-author attribution [130].

A similar-size group of papers have instead employed SHAP for explanation purposes. In the following, we report the task for which explainability was sought through SHAP. Upadhyay et al. proposed a new model for fake health news detection [131]. Abbruzzese et al. proposed a new architecture for OCR anomaly detection and correction [132]. Benedetto et al. proposed an architecture for emotional reaction prediction for social posts [133]. Rizinski et al. proposed a framework to perform a lexicon-based sentiment analysis [134]. Sageshima et al. proposed a method to classify donors with high-risk kidneys in [135].

Then came several papers that compared different explainability approaches, including LIME and SHAP, either alone or in combination. Most of them considered gradient-based approaches. El Zini et al. employed LIME, Anchors, and SHAP when introducing a new dataset to evaluate the performances of different models for sentiment analysis [136]. Lottridge et al. compared annotations from humans with respect to explanations provided by both LIME and Integrated Gradients within the scope of crisis alert identification [137]. Arashpour et al. compared a wide range of explainability methods falling into the classes of perturbation-based methods and gradient-based methods (Integrated Gradients, Gradient SHAP, Occlusion, the Fast Gradient Sign Method, Projected Gradient Descent, Minimal Perturbation, and Feature Ablation) for waste categorization in images [138]. Neely et al. compared LIME, Integrated Gradients, DeepLIFT, Grad-SHAP, and Deep-SHAP to measure their degree of concordance [139]. Komorowski et al. compared LIME, Attention Rollout, and LRP-Rollout for a model to detect COVID-19 based on X-ray images [82]. Thakur et al. used LIME and Grad-CAM in [111] to compare different models for plant disease identification from leaf images. Tornqvist et al. proposed integrating SHAP and Integrated Gradients for automatic short-answer grading (ASAG) [140]. Vareille et al. compared different explainability methods (SHAP, Grad-CAM, Integrated Gradients, Occlusion, and different variations of them dedicated to the task) for multivariate time series analysis [113]. Mishra et al. compared LIME, SHAP, and LRP in explaining models for hate speech detection [49]. For the same task, Yadav et al. compared LIME, SHAP, and Attention visualization in [98]. Malhotra and Jindal used SHAP and LIME in models for depressive and suicidal behavior detection [141]. Abdalla et al. employed LIME and SHAP when introducing a dataset to be used as a benchmark for human-written papers

classification [142]. Fiok et al. used both TreeSHAP (a variation of SHAP for tree-based models) and BertViz in [83].

Quite a smaller group of papers considered perturbation methods other than LIME or SHAP. Poulton et al. applied different methods (saliency, InputXGradient, Integrated Gradient, Occlusion, and GradientSHAP) to explain the decisions of transformers for the automatic short-answer grading task [115]. Kadir et al. compared Soundness Saliency and Grad-CAM with respect to image classification [112]. Tang et al. proposed a new method for explainability that consists of finding the relevance matrix that minimizes the difference between the loss computed on the original image and the perturbed version obtained by masking the image with the relevance matrix, applying the new technique to a new model for cancer survival analysis [143].

5.5. Hybrid

The methods described in Sections 5.1–5.4 fall sharply into one of the categories we identified at the beginning of Section 5. However, some methods have been proposed that employ a combination of approaches (in most cases, a couple). In this section, we describe those methods, devoting a subsection to each combination.

5.5.1. Activation + Attention

As we said before, Chefer et al. in [41] introduced a form of a relevance score that is the combination of the Attention Rollout method and LRP method with the assumption of an architecture based on self-attention. For the sake of brevity, we will call it LRP-Rollout. Indeed, for each transformer block B, they compute a score matrix $\overline{A}^B$, multiplying them (equally to what is performed in Attention Rollout) to obtain a global score matrix C:

$$C = \overline{A}^1 \cdot \overline{A}^2 \cdot \ldots \cdot \overline{A}^N \tag{19}$$

where $\overline{A}^1$ is the score matrix for the last transformer block and $\overline{A}^N$ is the score matrix for the first transformer block. The difference with respect to the Attention Rollout methods is the definition of each score matrix. They are defined as

$$\overline{A}^B = I + \mathbb{E}_h \left(\nabla A^B \odot R^B \right)^+ \tag{20}$$

where ∇A^B is the gradient of the attention matrix A^B; $\odot$ is the Hadamard product (element-wise product); R^B is the relevance scores obtained with LRP over the softmax layer of the transformer block B; $\mathbb{E}_h$ is the average operation over all the heads in the transformer block B; and $(\ldots)^+$ indicates only the positive values, leaving all the others as 0.

The number of applications of the combination of Activation and Attention is quite small.

Bianco et al. compared different architectures for food recognition, using LRP-Rollout for the explanation [144]. A combination of Attention Rollout with the activations of the last layer was proposed by Black et al. when detecting image similarities in the input of transformers [145]. Sun et al. proposed a method to estimate the relevance of each token in the input by masking all the attention matrices (in each layer) by the columns corresponding to the token we are computing, keeping also all the other values fixed in [146]. A comparison of LIME, Attention Rollout, and LRP-Rollout was carried out by Komorowski et al. in COVID-19 detection based on X-ray images [82].

5.5.2. Activation + Gradient

Gur et al. in [147] presented a method called Attribution-Guided Factorization (AGF), which combines the ideas behind both LRP and Grad-CAM. In their work, they tried to classify each neuron, with respect to the output class, as either in the background or in the foreground. The main idea behind the method is to use positive components of the gradients, which are then propagated back using variations of Relevance Propagation methods.

A similar approach is the basis of Softmax-Gradient Layer-wise Relevance Propagation (SGLRP), presented by Iwana et al. in [148]. They considered the relevance scores computed from the last layer, just as the gradients computed for the selected class, and then used classical LRP rules to propagate relevance to the input layer.

A method called FullGrad, thought for convolutional networks, was presented by Srinivas and Fleuret in [149]. They summed two terms: the InputXGradient term and the sum of the gradients with respect to each channel of each layer, each of them multiplied by its bias term. All the terms were rescaled and transformed according to the input image, just before the global sum. The result is a saliency map of the input image (with respect to the class of interest).

There also exists another method called Eigen-Grad-CAM, derived from both Eigen-CAM and Grad-CAM, that, as far as we know, has never been published, and it is only present in a Python library (https://github.com/jacobgil/pytorch-grad-cam (accessed on 1 March 2024)). The method consists of multiplying activations and gradients (like Grad-CAM) but then projecting on its first eigenvector (just like Eigen-CAM).

Ferrando et al. extended the ALTI method (see Section 5.1), multiplying each component $A^l_{i,j}V(h^{l-1}_j)$ by the matrix just before the final softmax computation in the network (supposing a next token prediction task) built by training and dropping the difference with respect to the module output. The method is called ALTI-logit. They also proposed to measure the difference between two different tokens in the output, subtracting the results of ALTI-logit for the two tokens. They called it Contrastive ALTI-logit. They also compared their methods with the Contrastive Gradient Norm and Contrastive InputXGradient in [54].

Hroub et al. compared different models for pneumonia and COVID-19 existence prediction from X-rays and used many methods (Grad-CAM, Grad-CAM++, Eigen-Grad-CAM, and AblationCAM) to produce saliency maps [57].

Arian et al. proposed a new architecture for human age and gender estimation from panoramic radiographs, using FullGrad to create saliency maps [150].

5.5.3. Activation + Perturbation

Sometimes, the difference in activations is measured when the input changes. The class of methods opting for this approach actually employs both techniques, perturbing the input and measuring the activation.

One of the main methods employing this approach is Deep Learning Important FeaTures (DeepLIFT), presented by Shrikumar et al. in [151]. It is based on the same ideas behind LRP, but it uses the differences in activations given by the input image and a perturbed one, which acts as the reference image, depending entirely on the task (for the MNIST dataset, they used a totally black image, while for DNA sequences, they used a reference input, which has expected frequencies for the base components in each position).

In the already cited work concerning SHAP, Lundberg et al. also introduced a modified version of DeepLIFT as an approximation of SHAP values [119]. The method consists of applying DeepLIFT propagation rules, defined in terms of SHAP values, recursively on small components of the input to obtain relevance scores for the input features.

Score-CAM, presented in [116], is a variation of Grad-CAM that does not use gradients but just the activations of the last convolutional layer, where each activation matrix for a filter is normalized, resized according to the original input size, and then passed again through the network to obtain the weights used to compute the linear combination of activation matrices, which will correspond to the saliency map.

Xie et al. in [152] presented ViT-CX, a technique dedicated to visual transformers. Considering the patches of the input image elaborated by the ViT, they used the output matrix of a module (usually the last one in the architecture) as the embeddings of the patches, building a new matrix composed of all the results of the same module for all the patches. All the slices of the matrix (each of them composed of all the patches) are then taken separately, upscaled according to the input image, and normalized between 0 and 1. All these slices become masks for the input. To reduce the cardinality of the problem, we

aggregate the masks using some clustering technique and then we add noise to each mask. We consider the saliency map from a mask as the output of the model after we apply the mask, plus the difference in the output between the original input and the original input with noise added. The final saliency map is the sum of all the saliency maps, with each pixel component normalized with respect to the number of times the pixel is in a mask.

The papers employing these hybrid techniques typically compare several approaches to explainability. Neely et al. compared LIME, Integrated Gradients, DeepLIFT, Grad-SHAP, and Deep-SHAP and evaluated their degree of concordance [139]. Englebert et al. presented a method called Transformer Input Sampling (TIS), which can be considered a variant of ViT-CX, but instead of masking the input image, they randomly picked a set of patches of the input and then computed the output for that set of embeddings. Furthermore, they compared TIS with ViT-CX, TAM, TMME, LRP-Rollout, Attention Rollout, BT, RISE, Integrated Gradients, and SmoothGrad [153]. Jourdan et al. proposed a method composed of three steps: (1) using Non-negative Matrix Factorization on the activation matrix from the last layer to obtain the topic matrix; (2) applying perturbation on the topic matrix; and (3) applying Occlusion to obtain the relevance scores [154].

5.5.4. Attention + Gradient

The method AttCAT, presented by Qiang et al. in [155], was inspired by Grad-CAM, but it was focused on transformers for text-related tasks. Considering the output of a single layer l of a transformer architecture as composed by the columns $h_1^l, \ldots, h_i^l, \ldots, h_n^l$, each of them representing the output for the i-th token, we can compute what they call Class Activation Tokens (CATs) for the i-th token in the l-th layer for the output class c as

$$\mathrm{CAT}_{i,l}^c = \frac{\partial y^c}{\partial h_i^l} \odot h_i^l \tag{21}$$

After computing the CAT values, we can compute the relevance of each token i with respect to the class c as

$$R_i^c = \sum_l \mathbb{E}_h \left(A_i^l \cdot \mathrm{CAT}_{i,l}^c \right) \tag{22}$$

A_i^l is the attention weight in the l-th layer for the i-th token.

As an example of this class of explainability techniques, we can visually compare different methods, including AttCAT, dedicated to NLP tasks, in Figure 8. In this type of visualization, the color gradient represents the importance of each token (in this case, each word) with respect to the classification of the sentence.

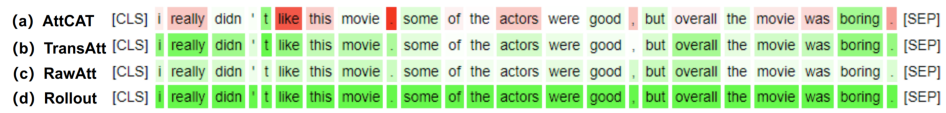

Figure 8. Example of visualization for methods devoted to NLP tasks. The green shade indicates an important positive impact whereas the red shade means otherwise. Darker colors represent higher impact scores [155].

Chefer et al. presented a method based both on attention weights and gradient computing in [156]. For the sake of brevity, we will call it TMME. It is compatible both with transformers with either text or image input and transformers with mixed input (text and image). Also, it can be applied both on self-attention-based architectures and co-attention-based architectures. Considering s and q, representing either a text or image, the relevancy maps are initialized as

$$R^{ss} = \mathbb{I} \tag{23}$$

$$R^{sq} = 0 \tag{24}$$

where $s \neq q$, and also considering the dimensions of the matrices compliant with the input features. The matrices are then updated recursively as

$$R^{ss} = R^{ss} + \overline{A} \cdot R^{ss} \tag{25}$$

$$R^{sq} = R^{sq} + \overline{A} \cdot R^{sq} \tag{26}$$

for self-attention layers, and it is updated recursively as

$$R^{sq} = R^{sq} + (R^{ss})^T \cdot \overline{A} \cdot R^{qq} \tag{27}$$

$$R^{ss} = R^{ss} + \overline{A} \cdot R^{qs} \tag{28}$$

for co-attention layers. In all previous equations, we refer to $\overline{A}$, which is formally defined as

$$\overline{A} = \mathbb{E}_h\Big((\nabla A \odot A)^+\Big) \tag{29}$$

where A represents the attention matrix of the layer, $\mathbb{E}_h$ is the average over all the heads in the layer, and the gradient is defined as

$$\nabla A = \frac{\partial Y_c}{\partial A} \tag{30}$$

A variant of TMME called Grad-SAS (Gradient-guided Self-Attention Summation) was proposed by Sun et al. in [157], where the relevance scores are just the sum of the $\overline{A}$ terms. another variant of TMME was proposed by Huang et al. in [158], where the updating rules for relevance values are modified as a weighted sum (with weights computing as a function of the gradient) for multi-modal models.

In [159], Liu et al. proposed a method to explain the classification of a text by a transformer-based model, composed of two methods: Attention Gradient (AGrad) and Relevance Propagation from Attention Gradient (RePAGrad). These methods return a relevance score for each token, which can be either positive or negative depending on the contribution given by the token for the classification. It computes the gradient of the loss to find if the contribution is positive (or negative), and it also backpropagates the relevance recursively using the Layer-wise Relevance Propagation (LRP) technique. Formally, the relevance score R_j for the j-th input token is defined as

$$R_j = \sum_i \left[R'_{i,j} \cdot \left(-\frac{\partial L}{\partial A_j} \cdot A_j \right) \right] \tag{31}$$

where $R'_{i,j}$ is the relevance score for the i-th contextual embedding with respect to the j-th input token computed by using the LRP method (this is true only for the RePAGrad method), A_i is the attention weight for the j-th input token, and L is the loss of the model. The variation in the AGrad method, with respect to RePAGrad, is the R' matrix, which is defined as equal to the identity matrix.

The product of the AGrad and RePAGrad scores was proposed by Thiruthuvaraj et al. in [160]. The combination of SHAP and Integrated Gradients was proposed by Tornqvist et al. for automatic short-answer grading (ASAG) [140].

A comparison of TMME and Grad-CAM was carried out by Wollek et al. for pneumothorax classification from chest radiographs [110]. Another comparison between Label Attribution (which is a variation of TMME) and Optimal Transport (the comparison between activations of different input types) was carried out by Ramesh and Koh in [56].

Sun et al. in [146] proposed a method to estimate the relevance of each token in the input by masking all the attention matrices (in each layer) by the columns corresponding to the token we are computing, keeping also all the other values fixed (we compute a forward pass on the original model, then we freeze the input of each layer, and we mask the attention matrices). The result of the last layer is extracted to represent the Shapley

values (so the relevance of each token). They also proposed two variations of Grad-CAM based on attention: the first one consists of computing the gradient of attention multiplied by the attention, and the second one is the same but expanded to the second order.

5.5.5. Attention + Perturbation

For this combination, we do not have significantly innovative methods.

However, Setzu et al. explained the output of a transformer-based model for many NLP tasks by extracting structured triples from the input text (for instance: subject, verb, and object), then perturbing the triples (using WordNet), and finally assigning a score to each perturbed triple based on attention weights with respect to the original input [161]. Correia et al. proposed a method for the explainability of face detection tasks by computing a binary attention matrix (using a threshold), using that matrix to compute a set of masks to perturb the image, and finally measuring the difference between the outputs of the original image and the perturbed one, repeating the procedure to obtain values to be merged to compose a relevance matrix [162].

5.5.6. Gradient + Perturbation

The most relevant method for this combination is the Integrated Gradients method, presented by Sundararaja et al. in [163]. They proposed to compute the integral of all the gradients computed in each point from a baseline input to the actual input. Formally, we consider a function $F : \mathbb{R}^n \to [0, 1]$ as the representation of our model (for a single output), x as the input, and x' as the baseline input (i.e., for images, it could be a black image, and for text, an embedding composed only by 0s). The relevance of each component i of the input is defined as

$$R_i = (x_i - x_i') \cdot \int_{\alpha=0}^{1} \frac{\partial F(x_i' + \alpha(x_i - x_i'))}{\partial x_i} d\alpha \tag{32}$$

The integral can be approximated by using the Riemann sum method as follows:

$$R_i = (x_i - x_i') \cdot \sum_{k=1}^{m} \frac{\partial F\left(x_i' + \frac{k}{m}(x_i - x_i')\right)}{\partial x_i} \cdot \frac{1}{m} \tag{33}$$

where m is the number of steps in the Riemann approximation. The authors experimented and found that a fair approximation is obtained using values in the (20,300) range for m. Chambon et al. employed Integrated Gradients for a transformer-based model to detect COVID-19 presence from radiology reports [164].

A variation of Integrated Gradients called Discretized Integrated Gradients was proposed by Sanyal and Ren in [165]. The variation appears to be more suitable for text embeddings. The difference with respect to the original technique lies in the interpolation between the points x and x', which is a linear interpolation in Integrated Gradients, while in Discretized Integrated Gradients, it is a path composed of embedding vectors for existing words, chosen each time according to some distance with respect to the previous vector. Another variation of Integrated Gradients, though never published, has been implemented in the `captum` (https://github.com/pytorch/captum (accessed on 1 March 2024)) library. It is called Layer Integrated Gradients and is indeed equal to Integrated Gradients, though applied on a specific layer (to be chosen by the user) instead of using just the global output and input.

There also exists a commonly used method called Gradient SHAP, developed by the same authors of SHAP with the aim of extending Integrated Gradients to estimate shapely values. Again, it has never been published, but it is implemented inside the Python `shap` library (https://github.com/shap/shap (accessed on 1 March 2024)).

Smilkov et al. in [166] presented SmoothGrad, a method to compute saliency maps for images. The relevances are computed as the average over all the gradients computed on a set of input images, each of them created as the original image plus some Gaussian noise.

Several papers employed a variety of explainability techniques including a combination of gradient-based and perturbation-based techniques. Lottridge et al. compared annotations by humans with explanations provided by both LIME and Integrated Gradients in crisis alert identification [137]. Cornia et al. applied three different gradient-based methods (saliency, Guided Backpropagation, and Integrated Gradients), aggregating the scores for different zones in the image [114]. Poulton et al. employed saliency, InputXGradient, Integrated Gradient, Occlusion, and GradientSHAP for automatic short-answer grading [115]. Arashpour et al. employed an even wider choice, including Integrated Gradients, Gradient SHAP, Occlusion, the Fast Gradient Sign Method, Projected Gradient Descent, Minimal Perturbation, and Feature Ablation for waste categorization in images [138]. Neely et al. compared LIME, Integrated Gradients, DeepLIFT, Grad-SHAP, and Deep-SHAP in [139]. Vareille et al. compared SHAP, Grad-CAM, Integrated Gradients, Occlusion, and different variations of them for a multivariate time series analysis [113]. Maladry et al. used LRP-Rollout, Discretized Integrated Gradients, and Layer Integrated Gradients when searching for bias in irony detection [167].

5.5.7. Attention + Gradient + Perturbation

This is the only combination of three techniques for which we have found relevant literature.

Yuan et al. presented a method called Transition Attention Maps (TAMs). This method considers the information flow inside the model as a Markov process, where the attention matrices represent the probabilities of state change. They accumulated attention matrices similarly to Attention Rollout, computing then the Integrated Gradients' relevance with respect to the last attention block and finally obtaining the relevance map as the product of the accumulated attention and the IG relevance map [168].

Chen et al. proposed a method (subsequently) called Bidirectional Transformers (BTs) that produces a saliency map by the element-wise multiplication of two matrices: the relevance matrix obtained by applying Integrated Gradients to the last attention block and a matrix obtained similarly to Attention Rollout but after rescaling the columns of each attention matrix by using rescaling factors obtained as a function of each module's output [169].

5.6. Overall Analysis

So far, we have described each technique according to the classification put forward at the beginning of Section 5. In this section, we analyze the relative extension and relevance of those techniques.

In Table 1, we report the major references for the techniques surveyed in Sections 5.1–5.5.7. The references listed in Table 1 concern those papers where the technique was proposed first. As we mentioned earlier, it may well happen that the technique was proposed first in one context and later applied to transformers, possibly with slight modifications.

The papers specifically dealing with transformers are instead shown in Table 2, again listed by the type of techniques they employ. The relative size of each category is shown in the heatmap of Table 3. Cells outside the main diagonal account for the combinations of two techniques. We see the dominance of attention-based techniques, followed by perturbation-based techniques. Hybrid methods are quite less represented.

Table 1. References for the initial proposal of major techniques.

Class	References
Response	LRP [38], Partial-LRP [40], RAP [42], CLRP [43], CAM [44], ALTI [45], Input Erasure [46], CAV [47], Eigen-CAM [48]
Attention	Attention Rollout [61], Attention Flow [61]
Gradient	Saliency [100], Guided Backpropagation [101], InputXGradient [102], Contrastive Gradient Norm [103], Contrastive InputXGradient [103], Grad-CAM [104], Grad-CAM++ [105]
Perturbation	Occlusion [117], LIME [118], SHAP [119], Anchors [121], RISE [122], Soundness Saliency [123], AblationCAM [124]
Response + Attention	LRP-Rollout [41]
Response + Gradient	AGF [147], SGLRP [148], FullGrad [149], Eigen-Grad-CAM
Response + Perturbation	DeepLIFT [151], DeepSHAP [119], Score-CAM [116], ViT-CX [152]
Attention + Gradient	AttCAT [155], TMME [156], AGrad [159], RePAGrad [159]
Gradient + Perturbation	Integrated Gradients [163], Discretized Integrated Gradients [165], Layer Integrated Gradients, Gradient SHAP, SmoothGrad [166]
Attention + Gradient + Perturbation	TAM [168], BT [169]

Table 2. Classification of papers on XAI for transformers.

Class	References
Response	[49–60]
Attention	[50,62–99]
Gradient	[57,58,81,95,106–115]
Perturbation	[49,82,83,98,111–113,115,125–143]
Response + Attention	[41,82,144–146]
Response + Gradient	[54,57,150]
Response + Perturbation	[139,153,154]
Attention + Gradient	[56,110,140,146,156–160]
Gradient + Perturbation	[113–115,137–139,164,167]
Attention + Perturbation	[161,162]

However, if we turn our attention to the importance rather than sheer size, as embodied by the number of citations, the situation changes. As can be seen in the heatmap of Table 4, the dominant class is a hybrid one, namely Activation+Attention techniques, which have nearly twice as many citations as sheer Attention-based techniques. Another hybrid technique, Attention+Gradient, ranks third. Anyway, all the top three techniques include attention, either alone or as one of the components.

Table 3. Number of papers by technique class.

	Attention	Gradient	Perturbation	Activation
Attention	38	10	2	5
Gradient	10	14	8	3
Perturbation	2	8	27	3
Activation	5	3	3	12

Table 4. Citations by technique class.

	Attention	Gradient	Perturbation	Activation
Attention	123	85	1	243
Gradient	85	57	26	1
Perturbation	1	26	51	2
Activation	243	1	2	42

Finally, if we look at the papers proposing methodological innovation in Table 5, we find attention-based and activation-based methods at the top among the single-class methods, but the joint use of gradient and attention is attracting more research efforts overall.

Table 5. Number of papers presenting new methods by category.

	Attention	Gradient	Perturbation	Activation
Attention	4	7	2	3
Gradient	7	2	1	1
Perturbation	2	1	2	2
Activation	3	1	2	5

6. Discussion and Conclusions

Our survey has highlighted an explosion of papers concerning the explainability of transformers. This trend accompanies the development of transformers' applications. As their domain of application expands into image analysis from the original NLP context, the development of explainability methods follows suit. As transformer models are increasingly applied to multi-modal tasks (involving text, images, and audio), research should extend to develop and assess explainability methods that can handle the intricacies of multiple data types. Understanding how transformers integrate and prioritize information across modes is essential for comprehensive explanations. Also, as transformers continue to grow in size, developing explainability methods that can scale efficiently with model complexity will be crucial. A trade-off will have to be sought between fast and computationally efficient explanations and depth or accuracy, especially for real-time applications.

The favorite class of methods, as measured by the number of papers, still appears to be attention-based. That would appear as a natural choice since the introduction of attention layers has been a major feature of transformers. However, the literature itself warns that attention is not a synonym for explainability. If we look at citations as a measure of interest instead of the sheer number of papers, methods based on attention only are overcome by methods based on both attention and activation, i.e., that look at the output of neurons tracing the relevance of features from the network output back to its input. Also, if we

just consider papers proposing methodological innovations, the joint use of gradient and attention is the topic appearing most in those papers.

Another relevant trend is the adoption of visualization tools to help explain the relevance of features. Though they do not introduce a real innovation in detecting the most relevant features, visual tools represent a step forward in the communication of explainability results to the layman or the professional in charge of using ML tools. In particular, developing interactive visualization tools that allow users to query, manipulate, and explore model decisions in real time could significantly enhance understanding. This is particularly true in the medical field, where ML tools are heavily used to support doctors in diagnosis tasks. Also, incorporating feedback from end-users, especially those without technical expertise, could guide the development of more user-friendly and accessible explainability methods. User studies and participatory design processes could help identify the most useful explanation approaches according to the context of usage.

Unfortunately, the diffusion of explainability methods and their introduction in ready-to-use libraries (e.g., in Python) may make it easier to apply such tools as black boxes without a clear awareness of their preferred field of application or limitations. In the long term, this behavior may lead to wrongly feeling self-confident and excessively relying on the method's output, though it may not be appropriate.

An additional concern is the relative lack of literature on evaluating explainability methods. There is a need for standardized, objective metrics to evaluate the effectiveness of explainability methods. These metrics should assess not just the accuracy of explanations but also their comprehensibility to humans. Developing benchmarks that can compare different explainability approaches on a common ground would enable more direct comparisons and facilitate progress. Incorporating knowledge from fields such as cognitive science, psychology, and philosophy could also offer new perspectives on what constitutes a good explanation and how to evaluate the explainability of AI models from a human-centric viewpoint, aiming at explainability methods that are deeply aligned with human cognition and understanding and also recognize ethical and fairness issues. This topic is certainly the most compelling to investigate if we wish to identify the most effective ones.

Author Contributions: Conceptualization, M.N. and P.F.; methodology, M.N. and P.F.; software, P.F.; validation, P.F. and M.N.; formal analysis, M.N.; investigation, P.F.; resources, P.F.; data curation, P.F.; writing—original draft preparation, M.N.; writing—review and editing, M.N.; visualization, P.F.; supervision, M.N.; project administration, M.N.; funding acquisition, M.N. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially funded by the European Union–Next Generation EU within the framework of PRIN 2022 Project "MEDICINE+AI, Law and Ethics for an Augmented and Human-Centered Medicine" (2022YB89EH)—CUP E53D23007020006.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AGF	Attribution-Guided Factorization
AGrad	Attention Gradient
ALTI	Aggregation of Layer-wise Token-to-Token Interactions
AttCATs	Attentive Class Activation Tokens
BERT	Bidirectional Encoder Representations from Transformers
BTs	Bidirectional Transformers
Ablation-CAM	Ablation Class Activation Mapping
CAM	Class Activation Mapping
CAVs	Concept Activation Vectors

CLRP	Contrastive Layer-wise Relevance Propagation
CLS	Cross-lingual summarization
CNN	Convolutional Neural Network
DeepLIFT	Deep Learning Important FeaTure
DistilBERT	Distilled BERT
DeepSHAP	Deep SHapley Additive exPlanations
DOAJ	Directory of open-access journals
Eigen-CAM	Eigenvalue Class Activation Mapping
Grad-CAM	Gradient weighted Class Activation Mapping
LIME	Local Interpretable Model-agnostic Explanation
LRP	Layer-wise Relevance Propagation
LRP-Rollout	Layer-wise Relevance Propagation Rollout
MDPI	Multi-disciplinary Digital Publishing Institute
ML	Machine Learning
NN	Neural network
Partial-LRP	Partial Layer-wise Relevance Propagation
RAP	Relative Attributing Propagation
RePAGrad	Relevance Propagation from Attention Gradient
RISE	Randomized Input Sampling for Explanation
RoBERTa	Robustly optimized BERT pretraining approach
Score-CAM	Score Class Activation Mapping
SGLRP	Softmax-Gradient Layer-wise Relevance Propagation
SHAP	SHapley Additive exPlanations
TAMs	Transition Attention Maps
TMME	Transformer Multi-Modal Explainability
ViT-CX	Vision Transformers Causal eXplanation

References

1. Islam, S.; Elmekki, H.; Elsebai, A.; Bentahar, J.; Drawel, N.; Rjoub, G.; Pedrycz, W. A comprehensive survey on applications of transformers for deep learning tasks. *Expert Syst. Appl.* **2023**, *241*, 122666. [CrossRef]
2. Parvaiz, A.; Khalid, M.A.; Zafar, R.; Ameer, H.; Ali, M.; Fraz, M.M. Vision Transformers in medical computer vision—A contemplative retrospection. *Eng. Appl. Artif. Intell.* **2023**, *122*, 106126. [CrossRef]
3. Karita, S.; Chen, N.; Hayashi, T.; Hori, T.; Inaguma, H.; Jiang, Z.; Someki, M.; Soplin, N.E.Y.; Yamamoto, R.; Wang, X.; et al. A comparative study on transformer vs rnn in speech applications. In Proceedings of the 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Singapore, 14–18 December 2019; pp. 449–456.
4. Ahmed, S.; Nielsen, I.E.; Tripathi, A.; Siddiqui, S.; Ramachandran, R.P.; Rasool, G. Transformers in time-series analysis: A tutorial. *Circuits Syst. Signal Process.* **2023**, *42*, 7433–7466. [CrossRef]
5. Thampi, A. *Interpretable AI: Building Explainable Machine Learning Systems*; Simon and Schuster: New York, NY, USA, 2022.
6. Marcinkevičs, R.; Vogt, J.E. Interpretable and explainable machine learning: A methods-centric overview with concrete examples. *WIREs Data Min. Knowl. Discov.* **2023**, *13*, e1493. [CrossRef]
7. Burkart, N.; Huber, M.F. A survey on the explainability of supervised machine learning. *J. Artif. Intell. Res.* **2021**, *70*, 245–317. [CrossRef]
8. Montavon, G.; Kauffmann, J.; Samek, W.; Müller, K.R. Explaining the predictions of unsupervised learning models. In Proceedings of the International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers, Vienna, Austria, 17 July 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 117–138.
9. Heuillet, A.; Couthouis, F.; Díaz-Rodríguez, N. Explainability in deep reinforcement learning. *Knowl.-Based Syst.* **2021**, *214*, 106685. [CrossRef]
10. Carvalho, D.V.; Pereira, E.M.; Cardoso, J.S. Machine learning interpretability: A survey on methods and metrics. *Electronics* **2019**, *8*, 832. [CrossRef]
11. Van Lent, M.; Fisher, W.; Mancuso, M. An explainable artificial intelligence system for small-unit tactical behavior. In Proceedings of the National Conference on Artificial Intelligence, San Jose, CA, USA, 25–29 July 2004; AAAI Press: Washington, DC, USA, 2004; pp. 900–907.
12. Bibal, A.; Lognoul, M.; De Streel, A.; Frénay, B. Legal requirements on explainability in machine learning. *Artif. Intell. Law* **2021**, *29*, 149–169. [CrossRef]
13. Waswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; Neural Information Processing Systems Foundation, Inc. (NeurIPS): San Diego, CA, USA, 2017.
14. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 6000–6010.

15. Lin, Z.; Feng, M.; Santos, C.N.d.; Yu, M.; Xiang, B.; Zhou, B.; Bengio, Y. A structured self-attentive sentence embedding. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
16. Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv* **2014**, arXiv:cs.CL/1409.0473.
17. Liu, P.J.; Saleh, M.; Pot, E.; Goodrich, B.; Sepassi, R.; Kaiser, L.; Shazeer, N. Generating Wikipedia by Summarizing Long Sequences. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018
18. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. *Improving Language Understanding by Generative Pre-Training;* Technical Report; OpenAI: San Francisco, CA, USA, 2018.
19. OpenAI. *GPT-4 Technical Report*; OpenAI: San Francisco, CA, USA, 2023.
20. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; Long and Short Papers; Association for Computational Linguistics: Kerrville, TX, USA, 2019; Volume 1, pp. 4171–4186. [CrossRef]
21. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
22. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2019** arXiv:1910.01108.
23. Child, R.; Gray, S.; Radford, A.; Sutskever, I. Generating Long Sequences with Sparse Transformers. *arXiv* **2019**, arXiv:1904.10509.
24. Katharopoulos, A.; Vyas, A.; Pappas, N.; Fleuret, F. Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention. In Proceedings of the 37th International Conference on Machine Learning, Virtual, 13–18 July 2020.
25. Guo, Q.; Qiu, X.; Xue, X.; Zhang, Z. Low-Rank and Locality Constrained Self-Attention for Sequence Modeling. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *27*, 2213–2222. [CrossRef]
26. Li, L.H.; Yatskar, M.; Yin, D.; Hsieh, C.J.; Chang, K.W. VisualBERT: A Simple and Performant Baseline for Vision and Language. *arXiv* **2019**, arXiv:1908.03557.
27. Alayrac, J.B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; Lenc, K.; Mensch, A.; Millican, K.; Reynolds, M.; et al. Flamingo: A visual language model for few-shot learning. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 23716–23736.
28. Gemini Team, Google. *Gemini: A Family of Highly Capable Multimodal Models;* Technical Report; Google: Mountain View, CA, USA, 2023.
29. Colliot, O. *Machine Learning for Brain Disorders*; Springer Nature: Heidelberg, Germany, 2023.
30. Zini, J.E.; Awad, M. On the Explainability of Natural Language Processing Deep Models. *ACM Comput. Surv.* **2022**, *55*, 1–31. [CrossRef]
31. Kashefi, R.; Barekatain, L.; Sabokrou, M.; Aghaeipoor, F. Explainability of Vision Transformers: A Comprehensive Review and New Perspectives. *arXiv* **2023**, arXiv:2311.06786.
32. Vijayakumar, S. Interpretability in Activation Space Analysis of Transformers: A Focused Survey. In Proceedings of the CIKM 2022 Workshops Co-Located with 31st ACM International Conference on Information and Knowledge Management (CIKM 2022), Atlanta, GA, USA, 17–21 October 2022.
33. Braşoveanu, A.M.P.; Andonie, R. Visualizing Transformers for NLP: A Brief Survey. In Proceedings of the 2020 24th International Conference Information Visualisation (IV), Melbourne, VIC, Australia, 7–11 September 2020; pp. 270–279.
34. Stassin, S.; Corduant, V.; Mahmoudi, S.A.; Siebert, X. Explainability and Evaluation of Vision Transformers: An In-Depth Experimental Study. *Electronics* **2023**, *13*, 175. [CrossRef]
35. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
36. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *Int. J. Surg.* **2021**, *88*, 105906. [CrossRef] [PubMed]
37. Tricco, A.C.; Lillie, E.; Zarin, W.; O'Brien, K.K.; Colquhoun, H.; Levac, D.; Moher, D.; Peters, M.D.; Horsley, T.; Weeks, L.; et al. PRISMA extension for scoping reviews (PRISMA-ScR): Checklist and explanation. *Ann. Intern. Med.* **2018**, *169*, 467–473. [CrossRef] [PubMed]
38. Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.R.; Samek, W. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PLoS ONE* **2015**, *10*, e0130140. [CrossRef] [PubMed]
39. Ding, Y.; Liu, Y.; Luan, H.; Sun, M. Visualizing and Understanding Neural Machine Translation. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 1150–1159.
40. Voita, E.; Talbot, D.; Moiseev, F.; Sennrich, R.; Titov, I. Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 5797–5808.
41. Chefer, H.; Gur, S.; Wolf, L. Transformer Interpretability Beyond Attention Visualization. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual, 19–25 June 2021; pp. 782–791.

42. Nam, W.J.; Gur, S.; Choi, J.; Wolf, L.; Lee, S.W. Relative Attributing Propagation: Interpreting the Comparative Contributions of Individual Units in Deep Neural Networks. *AAAI* **2020**, *34*, 2501–2508. [CrossRef]
43. Gu, J.; Yang, Y.; Tresp, V. Understanding Individual Decisions of CNNs via Contrastive Backpropagation. In Proceedings of the Computer Vision—ACCV 2018, Perth, Australia, 2–6 December 2018; Springer Nature Switzerland AG: Cham, Switzerland, 2019; pp. 119–134.
44. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
45. Ferrando, J.; Gállego, G.I.; Costa-jussà, M.R. Measuring the Mixing of Contextual Information in the Transformer. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Abu Dhabi, United Arab Emirates, 7–11 December 2022; pp. 8698–8714.
46. Li, J.; Monroe, W.; Jurafsky, D. Understanding Neural Networks through Representation Erasure. *arXiv* **2016**, arXiv:1612.08220.
47. Kim, B.; Wattenberg, M.; Gilmer, J.; Cai, C.; Wexler, J.; Viegas, F.; Sayres, R. Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). In Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; Volume 80, pp. 2668–2677.
48. Muhammad, M.B.; Yeasin, M. Eigen-CAM: Class Activation Map using Principal Components. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–7.
49. Mishra, R.; Yadav, A.; Shah, R.R.; Kumaraguru, P. Explaining Finetuned Transformers on Hate Speech Predictions Using Layerwise Relevance Propagation. In Proceedings of the Big Data and Artificial Intelligence, Delhi, India, 7–9 December 2023; Springer Nature: Cham, Switzerland, 2023; pp. 201–214.
50. Thorn Jakobsen, T.S.; Cabello, L.; Søgaard, A. Being Right for Whose Right Reasons? In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Toronto, ON, Canada, 9–14 July 2023; pp. 1033–1054.
51. Yu, L.; Xiang, W. X-Pruner: eXplainable Pruning for Vision Transformers. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 24355–24363.
52. Chan, A.; Schneider, M.; Körner, M. XAI for Early Crop Classification. In Proceedings of the 2023 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Pasadena, CA, USA, 16–21 July 2023; pp. 2657–2660.
53. Yang, Y.; Jiao, L.; Liu, F.; Liu, X.; Li, L.; Chen, P.; Yang, S. An Explainable Spatial–Frequency Multiscale Transformer for Remote Sensing Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–15. [CrossRef]
54. Ferrando, J.; Gállego, G.I.; Tsiamas, I.; Costa-Jussà, M.R. Explaining How Transformers Use Context to Build Predictions. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, Toronto, ON, Canada, 9–14 July 2023; Volume 1: Long Papersm, pp. 5486–5513.
55. Madsen, A.G.; Lehn-Schioler, W.T.; Jonsdottir, A.; Arnardottir, B.; Hansen, L.K. Concept-Based Explainability for an EEG Transformer Model. In Proceedings of the 2023 IEEE 33rd International Workshop on Machine Learning for Signal Processing (MLSP), Rome, Italy, 17–20 September 2023; pp. 1–6.
56. Ramesh, K.; Koh, Y.S. Investigation of Explainability Techniques for Multimodal Transformers. In *Communications in Computer and Information Science*; Communications in Computer and Information Science; Springer Nature: Singapore, 2022; pp. 90–98.
57. Hroub, N.A.; Alsannaa, A.N.; Alowaifeer, M.; Alfarraj, M.; Okafor, E. Explainable deep learning diagnostic system for prediction of lung disease from medical images. *Comput. Biol. Med.* **2024**, *170*, 108012. [CrossRef] [PubMed]
58. Alammar, J. Ecco: An Open Source Library for the Explainability of Transformer Language Models. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations, Online, 1–6 August 2021; pp. 249–257.
59. van Aken, B.; Winter, B.; Löser, A.; Gers, F.A. VisBERT: Hidden-State Visualizations for Transformers. In Proceedings of the Web Conference 2020, Taipei, Taiwan, 20–24 April 2020; pp. 207–211.
60. Gao, Y.; Wang, P.; Zeng, X.; Chen, L.; Mao, Y.; Wei, Z.; Li, M. Towards Explainable Table Interpretation Using Multi-view Explanations. In Proceedings of the 2023 IEEE 39th International Conference on Data Engineering (ICDE), Anaheim, CA, USA, 3–7 April 2023; pp. 1167–1179.
61. Abnar, S.; Zuidema, W. Quantifying Attention Flow in Transformers. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 4190–4197.
62. Renz, K.; Chitta, K.; Mercea, O.B.; Koepke, A.S.; Akata, Z.; Geiger, A. PlanT: Explainable Planning Transformers via Object-Level Representations. In Proceedings of the 6th Conference on Robot Learning, Auckland, New Zealand, 14–18 December 2022; Volume 205, pp. 459–470.
63. Feng, Q.; Yuan, J.; Emdad, F.B.; Hanna, K.; Hu, X.; He, Z. Can Attention Be Used to Explain EHR-Based Mortality Prediction Tasks: A Case Study on Hemorrhagic Stroke. In Proceedings of the 14th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, Houston TX USA, 3–6 September 2023; pp. 1–6.
64. Trisedya, B.D.; Salim, F.D.; Chan, J.; Spina, D.; Scholer, F.; Sanderson, M. i-Align: An interpretable knowledge graph alignment model. *Data Min. Knowl. Discov.* **2023**, *37*, 2494–2516. [CrossRef]
65. Graca, M.; Marques, D.; Santander-Jiménez, S.; Sousa, L.; Ilic, A. Interpreting High Order Epistasis Using Sparse Transformers. In Proceedings of the 8th ACM/IEEE International Conference on Connected Health: Applications, Systems and Engineering Technologies, Orlando, FL, USA, 21–23 June 2023; pp. 114–125.

66. Kim, B.H.; Deng, Z.; Yu, P.; Ganapathi, V. Can Current Explainability Help Provide References in Clinical Notes to Support Humans Annotate Medical Codes? In Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI), Abu Dhabi, United Arab Emirates, 7 December 2022; pp. 26–34.
67. Clauwaert, J.; Menschaert, G.; Waegeman, W. Explainability in transformer models for functional genomics. *Brief. Bioinform.* **2021**, *22*, bbab060. [CrossRef]
68. Sebbaq, H.; El Faddouli, N.E. MTBERT-Attention: An Explainable BERT Model based on Multi-Task Learning for Cognitive Text Classification. *Sci. Afr.* **2023**, *21*, e01799. [CrossRef]
69. Chen, H.; Zhou, K.; Jiang, Z.; Yeh, C.C.M.; Li, X.; Pan, M.; Zheng, Y.; Hu, X.; Yang, H. Probabilistic masked attention networks for explainable sequential recommendation. In Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, Macao, China, 19–25 August 2023.
70. Wantiez, A.; Qiu, T.; Matthes, S.; Shen, H. Scene Understanding for Autonomous Driving Using Visual Question Answering. In Proceedings of the 2023 International Joint Conference on Neural Networks (IJCNN), Gold Coast, Australia, 18–22 June 2023; pp. 1–7.
71. Ou, L.; Chang, Y.C.; Wang, Y.K.; Lin, C.T. Fuzzy Centered Explainable Network for Reinforcement Learning. *IEEE Trans. Fuzzy Syst.* **2024**, *32*, 203–213. [CrossRef]
72. Schwenke, L.; Atzmueller, M. Show me what you're looking for. *Int. Flairs Conf. Proc.* **2021**, *34*, 128399. [CrossRef]
73. Schwenke, L.; Atzmueller, M. Constructing Global Coherence Representations: Identifying Interpretability and Coherences of Transformer Attention in Time Series Data. In Proceedings of the 2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA), Porto, Portugal, 6–9 October 2021; pp. 1–12.
74. Bacco, L.; Cimino, A.; Dell'Orletta, F.; Merone, M. Extractive Summarization for Explainable Sentiment Analysis using Transformers. In Proceedings of the Sixth International Workshop on eXplainable SENTIment Mining and EmotioN deTection, Hersonissos, Greece, 7 June 2021.
75. Bacco, L.; Cimino, A.; Dell'Orletta, F.; Merone, M. Explainable Sentiment Analysis: A Hierarchical Transformer-Based Extractive Summarization Approach. *Electronics* **2021**, *10*, 2195. [CrossRef]
76. Humphreys, J.; Dam, H.K. An Explainable Deep Model for Defect Prediction. In Proceedings of the 2019 IEEE/ACM 7th International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering (RAISE), Montreal, QC, Canada, 28 May 2019; pp. 49–55.
77. Hickmann, M.L.; Wurzberger, F.; Hoxhalli, M.; Lochner, A.; Töllich, J.; Scherp, A. Analysis of GraphSum's Attention Weights to Improve the Explainability of Multi-Document Summarization. In Proceedings of the The 23rd International Conference on Information Integration and Web Intelligence, Linz, Austria, 29 November–1 December 2021; pp. 359–366.
78. Di Nardo, E.; Ciaramella, A. Tracking vision transformer with class and regression tokens. *Inf. Sci.* **2023**, *619*, 276–287. [CrossRef]
79. Cremer, J.; Medrano Sandonas, L.; Tkatchenko, A.; Clevert, D.A.; De Fabritiis, G. Equivariant Graph Neural Networks for Toxicity Prediction. *Chem. Res. Toxicol.* **2023**, *36*, 1561–1573. [CrossRef] [PubMed]
80. Pasquadibisceglie, V.; Appice, A.; Castellano, G.; Malerba, D. JARVIS: Joining Adversarial Training with Vision Transformers in Next-Activity Prediction. *IEEE Trans. Serv. Comput.* **2023**, *01*, 1–14. [CrossRef]
81. Neto, A.; Ferreira, S.; Libânio, D.; Dinis-Ribeiro, M.; Coimbra, M.; Cunha, A. Preliminary study of deep learning algorithms for metaplasia detection in upper gastrointestinal endoscopy. In Proceedings of the International Conference on Wireless Mobile Communication and Healthcare, Virtual Event, 30 November–2 December 2022; Springer Nature: Cham, Switzerland, 2023; pp. 34–50.
82. Komorowski, P.; Baniecki, H.; Biecek, P. Towards evaluating explanations of vision transformers for medical imaging. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Vancouver, BC, Canada, 18–22 June 2023; pp. 3726–3732.
83. Fiok, K.; Karwowski, W.; Gutierrez, E.; Wilamowski, M. Analysis of sentiment in tweets addressed to a single domain-specific Twitter account: Comparison of model performance and explainability of predictions. *Expert Syst. Appl.* **2021**, *186*, 115771. [CrossRef]
84. Tagarelli, A.; Simeri, A. Unsupervised law article mining based on deep pre-trained language representation models with application to the Italian civil code. *Artif. Intell. Law* **2022**, *30*, 417–473. [CrossRef]
85. Lal, V.; Ma, A.; Aflalo, E.; Howard, P.; Simoes, A.; Korat, D.; Pereg, O.; Singer, G.; Wasserblat, M. InterpreT: An Interactive Visualization Tool for Interpreting Transformers. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Online, 19–23 April 2021; pp. 135–142.
86. Dai, Y.; Jayaratne, M.; Jayatilleke, B. Explainable Personality Prediction Using Answers to Open-Ended Interview Questions. *Front. Psychol.* **2022**, *13*, 865841. [CrossRef] [PubMed]
87. Gaiger, K.; Barkan, O.; Tsipory-Samuel, S.; Koenigstein, N. Not All Memories Created Equal: Dynamic User Representations for Collaborative Filtering. *IEEE Access* **2023**, *11*, 34746–34763. [CrossRef]
88. Zeng, W.; Gautam, A.; Huson, D.H. MuLan-Methyl-multiple transformer-based language models for accurate DNA methylation prediction. *Gigascience* **2022**, *12*, giad054. [CrossRef] [PubMed]
89. Belainine, B.; Sadat, F.; Boukadoum, M. End-to-End Dialogue Generation Using a Single Encoder and a Decoder Cascade With a Multidimension Attention Mechanism. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 8482–8492. [CrossRef] [PubMed]

90. Ye, X.; Xiao, M.; Ning, Z.; Dai, W.; Cui, W.; Du, Y.; Zhou, Y. NEEDED: Introducing Hierarchical Transformer to Eye Diseases Diagnosis. In Proceedings of the 2023 SIAM International Conference on Data Mining (SDM), Minneapolis-St. Paul Twin Cities, MN, USA, 27–29 April 2023; pp. 667–675.
91. Kan, X.; Gu, A.A.C.; Cui, H.; Guo, Y.; Yang, C. Dynamic Brain Transformer with Multi-Level Attention for Functional Brain Network Analysis. In Proceedings of the 2023 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI), Pittsburgh, PA, USA, 15–18 October 2023; pp. 1–4.
92. Qu, G.; Orlichenko, A.; Wang, J.; Zhang, G.; Xiao, L.; Zhang, K.; Wilson, T.W.; Stephen, J.M.; Calhoun, V.D.; Wang, Y.P. Interpretable Cognitive Ability Prediction: A Comprehensive Gated Graph Transformer Framework for Analyzing Functional Brain Networks. *IEEE Trans. Med. Imaging* **2023**, *43*, 1568–1578. [CrossRef] [PubMed]
93. Sonth, A.; Sarkar, A.; Bhagat, H.; Abbott, L. Explainable Driver Activity Recognition Using Video Transformer in Highly Automated Vehicle. In Proceedings of the 2023 IEEE Intelligent Vehicles Symposium (IV), Anchorage, AK, USA, 4–7 June 2023; pp. 1–8.
94. Shih, J.-L.; Kashihara, A.; Chen, W.; Chen, W.; Ogata, H.; Baker, R.; Chang, B.; Dianati, S.; Madathil, J.; Yousef, A.M.F.; et al. (Eds.) Recommending Learning Actions Using Neural Network. In Proceedings of the 31st International Conference on Computers in Education, Matsue, Japan, 4–8 December 2023.
95. Wang, L.; Huang, J.; Xing, X.; Yang, G. *Hybrid Swin Deformable Attention U-Net for Medical Image Segmentation*; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2023.
96. Kim, J.; Kim, T.; Kim, J. Two-pathway spatiotemporal representation learning for extreme water temperature prediction. *Eng. Appl. Artif. Intell.* **2024**, *131*, 107718. [CrossRef]
97. Monteiro, N.R.C.; Oliveira, J.L.; Arrais, J.P. TAG-DTA: Binding-region-guided strategy to predict drug-target affinity using transformers. *Expert Syst. Appl.* **2024**, *238*, 122334. [CrossRef]
98. Yadav, S.; Kaushik, A.; McDaid, K. Understanding Interpretability: Explainable AI Approaches for Hate Speech Classifiers. In *Explainable Artificial Intelligence*; Springer Nature: Cham, Switzerland, 2023; pp. 47–70.
99. Ma, J.; Bai, Y.; Zhong, B.; Zhang, W.; Yao, T.; Mei, T. Visualizing and Understanding Patch Interactions in Vision Transformer. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**. [CrossRef] [PubMed]
100. Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. In Proceedings of the Workshop at International Conference on Learning Representations, Banff, AB, Canada, 14–16 April 2014.
101. Springenberg, J.T.; Dosovitskiy, A.; Brox, T.; Riedmiller, M. Striving for Simplicity: The All Convolutional Net. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015; Workshop Track Proceedings; Bengio, Y., LeCun, Y., Eds.; 2015.
102. Kindermans, P.J.; Schütt, K.; Müller, K.R.; Dähne, S. Investigating the influence of noise and distractors on the interpretation of neural networks. *arXiv* **2016**, arXiv:1611.07270.
103. Yin, K.; Neubig, G. Interpreting Language Models with Contrastive Explanations. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Abu Dhabi, United Arab Emirates, 7–11 December 2022; pp. 184–198.
104. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual explanations from deep networks via Gradient-based localization. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 618–626. [CrossRef]
105. Chattopadhay, A.; Sarkar, A.; Howlader, P.; Balasubramanian, V.N. Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 839–847.
106. Sobahi, N.; Atila, O.; Deniz, E.; Sengur, A.; Acharya, U.R. Explainable COVID-19 detection using fractal dimension and vision transformer with Grad-CAM on cough sounds. *Biocybern. Biomed. Eng.* **2022**, *42*, 1066–1080. [CrossRef] [PubMed]
107. Thon, P.L.; Than, J.C.M.; Kassim, R.M.; Yunus, A.; Noor, N.M.; Then, P. Explainable COVID-19 Three Classes Severity Classification Using Chest X-Ray Images. In Proceedings of the 2022 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES), Kuala Lumpur, Malaysia, 7–9 December 2022; pp. 312–317.
108. Vaid, A.; Jiang, J.; Sawant, A.; Lerakis, S.; Argulian, E.; Ahuja, Y.; Lampert, J.; Charney, A.; Greenspan, H.; Narula, J.; et al. A foundational vision transformer improves diagnostic performance for electrocardiograms. *NPJ Digit. Med.* **2023**, *6*, 108. [CrossRef] [PubMed]
109. Wang, J.; Qu, A.; Wang, Q.; Zhao, Q.; Liu, J.; Wu, Q. TT-Net: Tensorized Transformer Network for 3D medical image segmentation. *Comput. Med. Imaging Graph.* **2023**, *107*, 102234. [CrossRef] [PubMed]
110. Wollek, A.; Graf, R.; Čečatka, S.; Fink, N.; Willem, T.; Sabel, B.O.; Lasser, T. Attention-based Saliency Maps Improve Interpretability of Pneumothorax Classification. *Radiol. Artif. Intell.* **2023**, *5*, e220187. [CrossRef] [PubMed]
111. Thakur, P.S.; Chaturvedi, S.; Khanna, P.; Sheorey, T.; Ojha, A. Vision transformer meets convolutional neural network for plant disease classification. *Ecol. Inform.* **2023**, *77*, 102245. [CrossRef]
112. Kadir, M.A.; Addluri, G.; Sonntag, D. Harmonizing Feature Attributions Across Deep Learning Architectures: Enhancing Interpretability and Consistency. In Proceedings of the KI 2023: Advances in Artificial Intelligence, Berlin, Germany, 26–29 September 2023; Springer Nature: Cham, Switzerland, 2023; pp. 90–97.

113. Vareille, E.; Abbas, A.; Linardi, M.; Christophides, V. Evaluating Explanation Methods of Multivariate Time Series Classification through Causal Lenses. In Proceedings of the 2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA), Thessaloniki, Greece, 9–13 October 2023; pp. 1–10.
114. Cornia, M.; Baraldi, L.; Cucchiara, R. Explaining transformer-based image captioning models: An empirical analysis. *AI Commun.* **2022**, *35*, 111–129. [CrossRef]
115. Poulton, A.; Eliens, S. Explaining transformer-based models for automatic short answer grading. In Proceedings of the 5th International Conference on Digital Technology in Education, Busan, Republic of Korea, 15–17 September 2021; pp. 110–116.
116. Wang, H.; Wang, Z.; Du, M.; Yang, F.; Zhang, Z.; Ding, S.; Mardziel, P.; Hu, X. Score-CAM: Score-Weighted Visual Explanations for Convolutional Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Virtual, 14–19 June 2020; pp. 111–119.
117. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. In Proceedings of the Computer Vision—ECCV 2014, Zurich, Switzerland, 6–12 September 2014; Springer International Publishing: Cham, Switzerland, 2014; pp. 818–833.
118. Ribeiro, M.T.; Singh, S.; Guestrin, C. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1135–1144.
119. Lundberg, S.M.; Lee, S.I. A unified approach to interpreting model predictions. In Proceedings ofthe 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 4768–4777.
120. Shapley, L.S. A Value for n-person Games. In *Contributions to the Theory of Games (AM-28), Volume II*; Harold William Kuhn, A.W.T., Ed.; Princeton University Press: Princeton, NJ, USA, 1953; pp. 307–318.
121. Ribeiro, M.T.; Singh, S.; Guestrin, C. Anchors: High-Precision Model-Agnostic Explanations. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
122. Petsiuk, V.; Das, A.; Saenko, K. RISE: Randomized input sampling for explanation of black-box models. *arXiv* **2018**, arXiv:1806.07421.
123. Gupta, A.; Saunshi, N.; Yu, D.; Lyu, K.; Arora, S. New definitions and evaluations for saliency methods: Staying intrinsic, complete and sound. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 33120–33133.
124. Desai, S.; Ramaswamy, H.G. Ablation-CAM: Visual Explanations for Deep Convolutional Network via Gradient-free Localization. In Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), Snowmass Village, CO, USA, 1–5 March 2020; pp. 972–980.
125. Mehta, H.; Passi, K. Social Media Hate Speech Detection Using Explainable Artificial Intelligence (XAI). *Algorithms* **2022**, *15*, 291. [CrossRef]
126. Rodrigues, A.C.; Marcacini, R.M. Sentence Similarity Recognition in Portuguese from Multiple Embedding Models. In Proceedings of the 2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA), Nassau, Bahamas, 12–14 December 2022; pp. 154–159.
127. Janssens, B.; Schetgen, L.; Bogaert, M.; Meire, M.; Van den Poel, D. 360 Degrees rumor detection: When explanations got some explaining to do. *Eur. J. Oper. Res.* 2023, *in press*. [CrossRef]
128. Chen, H.; Cohen, E.; Wilson, D.; Alfred, M. A Machine Learning Approach with Human-AI Collaboration for Automated Classification of Patient Safety Event Reports: Algorithm Development and Validation Study. *JMIR Hum. Factors* **2024**, *11*, e53378. [CrossRef] [PubMed]
129. Collini, E.; Nesi, P.; Pantaleo, G. Reputation assessment and visitor arrival forecasts for data driven tourism attractions assessment. *Online Soc. Netw. Media* **2023**, *37–38*, 100274. [CrossRef]
130. Litvak, M.; Rabaev, I.; Campos, R.; Campos, R.; Campos, R.; Jorge, A.M.; Jorge, A.M.; Jatowt, A. What if ChatGPT Wrote the Abstract?—Explainable Multi-Authorship Attribution with a Data Augmentation Strategy. In Proceedings of the IACT'23 Workshop, Taipei, Taiwan, 27 July 2023; pp. 38–48.
131. Upadhyay, R.; Pasi, G.; Viviani, M. Leveraging Socio-contextual Information in BERT for Fake Health News Detection in Social Media. In Proceedings of the 3rd International Workshop on Open Challenges in Online Social Networks, Rome, Italy, 4 September 2023; pp. 38–46.
132. Abbruzzese, R.; Alfano, D.; Lombardi, A. REMOAC: A retroactive explainable method for OCR anomalies correction in legal domain. In *Frontiers in Artificial Intelligence and Applications*; IOS Press: Amsterdam, The Netherlands, 2023.
133. Benedetto, I.; La Quatra, M.; Cagliero, L.; Vassio, L.; Trevisan, M. Transformer-based Prediction of Emotional Reactions to Online Social Network Posts. In Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, Toronto, ON, Canada, 14 July 2023; pp. 354–364.
134. Rizinski, M.; Peshov, H.; Mishev, K.; Jovanovik, M.; Trajanov, D. Sentiment Analysis in Finance: From Transformers Back to eXplainable Lexicons (XLex). *IEEE Access* **2024**, *12*, 7170–7198. [CrossRef]
135. Sageshima, J.; Than, P.; Goussous, N.; Mineyev, N.; Perez, R. Prediction of High-Risk Donors for Kidney Discard and Nonrecovery Using Structured Donor Characteristics and Unstructured Donor Narratives. *JAMA Surg.* **2024**, *159*, 60–68. [CrossRef]
136. El Zini, J.; Mansour, M.; Mousi, B.; Awad, M. On the evaluation of the plausibility and faithfulness of sentiment analysis explanations. In *IFIP Advances in Information and Communication Technology*; Springer International Publishing: Cham, Switzerland, 2022; pp. 338–349.

137. Lottridge, S.; Woolf, S.; Young, M.; Jafari, A.; Ormerod, C. The use of annotations to explain labels: Comparing results from a human-rater approach to a deep learning approach. *J. Comput. Assist. Learn.* **2023**, *39*, 787–803. [CrossRef]
138. Arashpour, M. AI explainability framework for environmental management research. *J. Environ. Manag.* **2023**, *342*, 118149. [CrossRef] [PubMed]
139. Neely, M.; Schouten, S.F.; Bleeker, M.; Lucic, A. A song of (dis)agreement: Evaluating the evaluation of explainable artificial intelligence in natural language processing. In *HHAI2022: Augmenting Human Intellect*; Frontiers in Artificial Intelligence and Applications; IOS Press: Amsterdam, The Netherlands, 2022.
140. Tornqvist, M.; Mahamud, M.; Mendez Guzman, E.; Farazouli, A. ExASAG: Explainable Framework for Automatic Short Answer Grading. In Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), Toronto, ON Canada, 13 July 2023; pp. 361–371.
141. Malhotra, A.; Jindal, R. XAI Transformer based Approach for Interpreting Depressed and Suicidal User Behavior on Online Social Networks. *Cogn. Syst. Res.* **2024**, *84*, 101186. [CrossRef]
142. Abdalla, M.H.I.; Malberg, S.; Dementieva, D.; Mosca, E.; Groh, G. A Benchmark Dataset to Distinguish Human-Written and Machine-Generated Scientific Papers. *Information* **2023**, *14*, 522. [CrossRef]
143. Tang, Z.; Liu, L.; Shen, Y.; Chen, Z.; Ma, G.; Dong, J.; Sun, X.; Zhang, X.; Li, C.; Zheng, Q.; et al. Explainable survival analysis with uncertainty using convolution-involved vision transformer. *Comput. Med. Imaging Graph.* **2023**, *110*, 102302. [CrossRef] [PubMed]
144. Bianco, S.; Buzzelli, M.; Chiriaco, G.; Napoletano, P.; Piccoli, F. Food Recognition with Visual Transformers. In Proceedings of the 2023 IEEE 13th International Conference on Consumer Electronics–Berlin (ICCE-Berlin), Berlin, Germany, 2–5 September 2023; pp. 82–87.
145. Black, S.; Stylianou, A.; Pless, R.; Souvenir, R. Visualizing Paired Image Similarity in Transformer Networks. In Proceedings of the 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–8 January 2022; pp. 1534–1543.
146. Sun, T.; Chen, H.; Qiu, Y.; Zhao, C. Efficient Shapley Values Calculation for Transformer Explainability. In Proceedings of the Pattern Recognition. Springer Nature Switzerland, Tepic, Mexico, 21–24 June 2023; pp. 54–67.
147. Gur, S.; Ali, A.; Wolf, L. Visualization of Supervised and Self-Supervised Neural Networks via Attribution Guided Factorization. *AAAI* **2021**, *35*, 11545–11554. [CrossRef]
148. Iwana, B.K.; Kuroki, R.; Uchida, S. Explaining Convolutional Neural Networks using Softmax Gradient Layer-wise Relevance Propagation. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Republic of Korea, 27–28 October 2019; pp. 4176–4185.
149. Srinivas, S.; Fleuret, F. Full-gradient representation for neural network visualization. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Curran Associates Inc.: Red Hook, NY, USA, 2019; pp. 4124–4133.
150. Arian, M.S.H.; Rakib, M.T.A.; Ali, S.; Ahmed, S.; Farook, T.H.; Mohammed, N.; Dudley, J. Pseudo labelling workflow, margin losses, hard triplet mining, and PENViT backbone for explainable age and biological gender estimation using dental panoramic radiographs. *SN Appl. Sci.* **2023**, *5*, 279. [CrossRef]
151. Shrikumar, A.; Greenside, P.; Kundaje, A. Learning important features through propagating activation differences. In Proceedings of the 34th International Conference on Machine Learning, Sydney, NSW, Australia, 6–11 August 2017; Volume 70, pp. 3145–3153.
152. Xie, W.; Li, X.H.; Cao, C.C.; Zhang, N.L. ViT-CX: Causal Explanation of Vision Transformers. In Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, Macao, China, 19–25 August 2023.
153. Englebert, A.; Stassin, S.; Nanfack, G.; Mahmoudi, S.; Siebert, X.; Cornu, O.; Vleeschouwer, C. Explaining through Transformer Input Sampling. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Paris, France, 2–3 October 2023; pp. 806–815.
154. Jourdan, F.; Picard, A.; Fel, T.; Risser, L.; Loubes, J.M.; Asher, N. COCKATIEL: COntinuous Concept ranKed ATtribution with Interpretable ELements for explaining neural net classifiers on NLP. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2023, Toronto, ON, Canada, 9–14 July 2023; pp. 5120–5136.
155. Qiang, Y.; Pan, D.; Li, C.; Li, X.; Jang, R.; Zhu, D. AttCAT: Explaining Transformers via attentive class activation tokens. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 5052–5064.
156. Chefer, H.; Gur, S.; Wolf, L. Generic Attention-model Explainability for Interpreting Bi-Modal and Encoder-Decoder Transformers. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 387–396.
157. Sun, T.; Chen, H.; Hu, G.; He, L.; Zhao, C. Explainability of Speech Recognition Transformers via Gradient-based Attention Visualization. *IEEE Trans. Multimed.* **2023**, *26*, 1395–1406. [CrossRef]
158. Huang, Y.; Jia, A.; Zhang, X.; Zhang, J. Generic Attention-model Explainability by Weighted Relevance Accumulation. In Proceedings of the 5th ACM International Conference on Multimedia in Asia, Taiwan, China, 6–8 December 2024; pp. 1–7.
159. Liu, S.; Le, F.; Chakraborty, S.; Abdelzaher, T. On exploring attention-based explanation for transformer models in text classification. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA 15–18 December 2021; pp. 1193–1203.

160. Thiruthuvaraj, R.; Jo, A.A.; Raj, E.D. Explainability to Business: Demystify Transformer Models with Attention-based Explanations. In Proceedings of the 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 4–6 May 2023; pp. 680–686.
161. Setzu, M.; Monreale, A.; Minervini, P. TRIPLEx: Triple Extraction for Explanation. In Proceedings of the 2021 IEEE Third International Conference on Cognitive Machine Intelligence (CogMI), Virtual, 13–15 December 2021; pp. 44–53.
162. Correia, R.; Correia, P.; Pereira, F. Face Verification Explainability Heatmap Generation. In Proceedings of the 2023 International Conference of the Biometrics Special Interest Group (BIOSIG), Darmstadt, Germany, 20–22 September 2023; pp. 1–5.
163. Sundararajan, M.; Taly, A.; Yan, Q. Axiomatic attribution for deep networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Volume 70; pp. 3319–3328.
164. Chambon, P.; Cook, T.S.; Langlotz, C.P. Improved Fine-Tuning of In-Domain Transformer Model for Inferring COVID-19 Presence in Multi-Institutional Radiology Reports. *J. Digit. Imaging* **2023**, *36*, 164–177. [CrossRef] [PubMed]
165. Sanyal, S.; Ren, X. Discretized Integrated Gradients for Explaining Language Models. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Online/Punta Cana, Dominican Republic, 7–11 November 2021; pp. 10285–10299.
166. Smilkov, D.; Thorat, N.; Kim, B.; Viégas, F.; Wattenberg, M. SmoothGrad: Removing noise by adding noise. *arXiv* **2017**, arXiv:1706.03825.
167. Maladry, A.; Lefever, E.; Van Hee, C.; Hoste, V. A Fine Line Between Irony and Sincerity: Identifying Bias in Transformer Models for Irony Detection. In Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, Toronto, ON, Canada, 14 July 2023; pp. 315–324.
168. Yuan, T.; Li, X.; Xiong, H.; Cao, H.; Dou, D. Explaining Information Flow Inside Vision Transformers Using Markov Chain. In Proceedings of the XAI 4 Debugging Workshop, Virtual, 14 December 2021.
169. Chen, J.; Li, X.; Yu, L.; Dou, D.; Xiong, H. Beyond Intuition: Rethinking Token Attributions inside Transformers. *Trans. Mach. Learn. Res.* **2023**, *2023*, 1–27.