Explainability in image captioning based on the latent space

Sofiane Elguendouze, Adel Hafiane, Marcilio C.P. de Souto, Anaïs Halftermeyer

Explainability in Image Captioning based on the Latent Space

Sofiane Elguendouze[1], Adel Hafiane[2], Marcilio C. P. de Souto[1], Anaïs Halftermeyer[1]

France

Abstract

This paper focuses on the representation/latent space in neural architectures to develop an end-to-end explanation approach for Image Captioning (IC) models. By injecting Gaussian perturbations into the latent space of each component of the architecture, we first analyze and identify the parts of the model likely to be the most decisive/influential in the caption generation. The results show that the visual part, mainly composed of visual encoding and attention mechanism, is more decisive than the language part, which could lead to more subtle explanations. We then follow this approach with an in-depth explanation protocol that also utilizes the latent space and focuses on the visual modality to design and compare two explanation methods with different scopes; (1) a surrogate-based method with Local Interpretable Model-Agnostic Explanations (LIME), with local scope. (2) a backpropagation-based method with Layer-wise Relevance Propagation (LRP) for global explanations. To assess the quality of the obtained explanations, we propose the new concept of Latent Ablation, which proves to be more consistent than classical Ablation, which usually leads to inconsistencies and truncated information. Extensive experiments show that both methods achieve comparable results and that their scope has no explicit impact on the quality of the explanations.

Keywords: Explainable Artificial Intelligence (XAI), Image Captioning,

[1]LIFO EA 4022, University of Orléans - INSA CVL, Orléans
[2]PRISME EA 4229, INSA CVL - University of Orléans, Bourges

Interpretability, Perturbation, Representation Space.

1. Introduction

Image captioning (IC) is one of the Vision to Language tasks aiming to generate captions (textual descriptions) for images. Most IC models are designed under the Encoder-Decoder framework, with typically a Convolutional Neural Network (CNN) as the encoder and a Recurrent Neural Network (RNN) as the decoder. Visual features are extracted from the image using the encoder, then transformed/translated at each time step by the decoder into textual data (words) constituting the output caption. An attention mechanism is often inserted between the two previous components giving a more sophisticated architecture (Encoder-Attention-Decoder). Basically, its main role is to guide the model at each decoding step so as to focus only on relevant information (regions) within the image, which results in more accurate captions.

Despite their high performance, the inner functioning of these architectures, mainly based on deep neural networks (DNN), means that they are considered to be black-boxes, making the decision process during the prediction of the caption difficult to understand. At present and to the best of our knowledge, the concern of explainability is not sufficiently studied in IC domain. Very few works have been proposed in the literature, most of which seek to establish a perceptive causality relationship between the output and the input as a visual saliency explanation. The authors in [10] generate a kind of word-region link (groundings), where the regions (concepts/objects) in the image are weighted and then highlighted according to their importance in the prediction of a target word in the caption. In [35], the authors adapt a decomposition method called Layer-wise Relevance Propagation (LRP)[3] for IC models with a CNN encoder, to generate heatmaps that reflect the importance of pixels in the input image in the prediction of each word in the caption.

A major problem of saliency explanation techniques is that, when used alone, they may be insufficient since they do not unblack-box the model [37] or its

inner components. Moreover, we cannot ascertain whether decisions made by the model, for example in the case of IC, are exclusively based on the foreground objects detected in the scene (by the aforementioned explanation methods) or whether background objects are involved in the captioning process [37].

A set of methods called "Signal-based explanation" offers the possibility for more scrutiny, by investigating black-box models at a deeper level, such as feature maps [49] and ablations [25]. Despite their ability to discern the important features considered by the model in the input while predicting the output, the explanations provided by these methods have proved to be largely distorted [37], which is caused by the ablation mechanism for example. Moreover, just like all other explanation methods, Signal-based methods do not address a fundamental question in explainability: are all components within the model of equal importance? Do they differ in terms of influence on the captioning process? To address this concern, we proposed, in a previous study [8], an explanation approach based on Gaussian perturbation of the representation space, which was shown to be more powerful than operating on the original input space (images) through its ability to identify more subtle explanatory elements and reveal the hidden evidence used by the model in the decision making. The idea was to determine the influence of each component of the captioning architecture on the captioning process. The results showed that the visual part is more involved when generating the caption than the language part, especially when it comes to the generation of contextual tags in the caption such as objects, which could lead to more fine-grained explanations for further interpretability.

Based on the distinctive strengths that have been noticed in the representation space, we propose in this paper to enlarge our approach by first extending the component influence study to a new dataset to reinforce our previous findings. We then exploit the latent space on two popular attribution-based explanatory approaches: LRP which is a method based on the backpropagation principle, and LIME [30] (Local Interpretable Model-Agnostic Explanations)

which is a surrogate[3]-based method. The global approach is illustrated in Fig. 1 as an end-to-end explanation framework. We call attribution methods, the set of methods operating in post-hoc explainability mode, and assigning relevance scores to inputs according to their importance for the prediction of outputs. LRP and LIME traditionally operate on the original input space, but we propose here to switch to the latent space to further emphasize its role in generating finer-grained explanations. Yet another difference between the two methods lies in their scope. While LRP derives an explanation based on the entire logic of the model and including the learned internal weights, which yields its global scope, LIME operates locally around each instance using only the inputs and the predicted outputs. It is therefore worthwhile analyzing the explanations provided by each of the two explanation methods to see if the scope of a method (local/global) has any effect on the quality of the explanations it provides.

The main contributions of the paper are as follows:

- In light of the promising results obtained on the MSCOCO2017[18] captioning dataset, we extend our method based on the (latent) representation space perturbation principle to a new dataset called Flickr30k[48] to strengthen our findings. The method seeks to isolate and identify the influence of each component that belongs to the captioning architecture. Perturbation has been shown to be more accurate than conventional ablation when used for generating explanations, where information is gradually modified rather than truncated. The representation space in addition has greater power to reveal hidden evidence used by the model while taking decisions. Our results reveal that the visual part of captioning models is more impactful and decisive than the language part, which means that this component represents a central element in further explainability steps.

- Based on LRP, we develop an adapted version called BU-LRP that gen-

[3] "A surrogate model is a simple model used to explain a complex model"[1].

erates explanations for Bottom-Up (BU)[4] IC architectures. The method emphasizes the role of the visual modality and operates completely on the representation space given its ability to bring out the latent clues used by the model to make decisions. So far as we know, the concept of adapting LRP attributions to the BU features that we propose here is entirely novel. Its interest is even more pronounced since most captioning architectures are typically based on BU features extracted using a Faster-RCNN [29] encoder rather than on the classical global CNN features.

- We design a Surrogate-based explanation approach called BU-LIME for BU captioning models to generate visual explanations. Instead of the classical intrinsic perturbation (blur/black) of the input data for LIME as in [32], this is the first work to introduce an intrinsic perturbation on the latent space to build the linear surrogate model. We propose two versions of LIME: the first is based on visual features perturbation and the second is based on full object perturbation, in order to study the impact of full conceptual units (the whole set of BU features concerning the target object) on the generation of explanations versus individual visual feature units. The resulting explanations are compared to those of the previous method (BU-LRP) in terms of correctness and agreement at both qualitative and quantitative scales.

- We introduce the new concept of *Latent Ablation* which also operates on the latent space to assess the quality of explanations, showing better consistency than the classical concept of ablation. To the best of our knowledge, a comparative study between attribution-based explanation methods has never been performed before. Therefore, it is interesting to compare the two distinct explanation techniques, backpropagation and surrogates, on the same image captioning task. This helps determine whether backpropagation offers improvements over linear model building in the context

[4]Bottom-Up image features correspond to the candidate regions from object detection.

of explainability, as traditionally observed for learning models.

The paper is organized as follows. Section 2 provides an overview of the fields of image captioning and explainable artificial intelligence, and then reviews the few existing approaches to explainable IC. Section 3 describes the proposed methods of explainability. Experimental results are shown and discussed in Sec. 4.

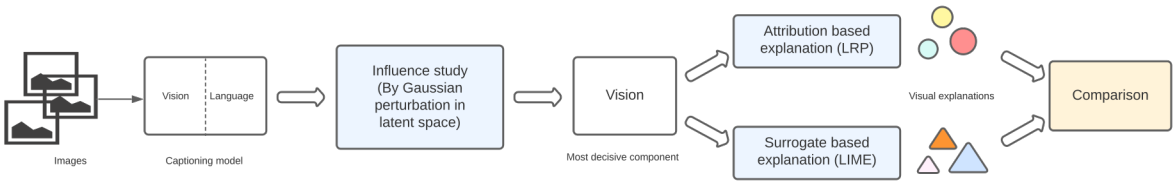

Figure 1: Overview of the proposed end-to-end explanation framework.

2. Related Work

2.1. Image captioning

Inspired by machine translation in natural language processing, the IC task is considered as the transfer from the visual modality (image) to the linguistic modality (caption). The most widely adopted architecture is the Encoder-Decoder[34], where an intermediate latent representation is obtained from the image using the encoder, then transformed by the decoder to obtain a sequence describing the input image. Based on the fact that looking at all details of the image at each decoding step is not efficient since the regions may vary in terms of importance, many attention mechanisms have been proposed to guide the model to focus only on relevant content while generating each word of the caption. These mechanisms include but are not limited to semantic attention [47], adaptive attention [23], X-linear attention [26] and attention on attention [13]. The progress made by transformer-based architectures [38] has motivated a number of recent works to replace the RNN decoder in captioning models with a transformer decoder [12, 15]. These complex architectures did not provide any

significant improvements in terms of caption quality. However, the use of transformers in both the encoder and decoder parts gave better results compared to replacing the decoding part only, as in [22].

Focusing on the relationships between objects detected in the image constitutes a different branch of research in IC, which could lead to finer-grained captions. [52] proposed a method based on semantic representation using Scene-graph decomposition. Scene-graph was previously proposed by [50]. It aims at building a semantic graphical representation of scene components, where nodes represent objects and edges represent predicates (relationships between pairs of objects, attributes such as colors, adjectives, etc.). The model relies on a Graph Convolutional Network (GCN) that aggregates neighborhood information in the graph and enriches the representation with contextual information that binds objects and predicates.

[20] designed a specialized captioning method for Chinese based on visual attention and topic modeling. By leveraging the strengths of visual attention to understand the details of the image on the one hand, and the non-negative matrix factorization (NMF) topic model to include topic textual information to guide the caption generation on the other hand, their method has proven to be effective in generating more diverse and accurate sentences. Driven by the lack of image information and the deviation of the generated captions from the main content of the image, [21] took captioning a step further by incorporating image labels from the convolutional network into the decoding model and proceeding with a dual attention mechanism consisting of visual attention on image features, and textual attention on the aforementioned textual labels, whose role is to increase information integrity in the generated caption.

In view of all these rich and complex architectures, the need for explainability has become increasingly important, giving rise to several types of explanation approaches.

2.2. Explainable Artificial Intelligence

Explainable Artificial Intelligence (XAI) is a new branch of research that has been gaining momentum lately, as the high complexity found in artificial intelligence (AI) models has led to interpretability problems that prevent humans from understanding the reasons for the decisions these models make. A wide range of interpretability methods has then been developed to provide clues to explain those decisions. These methods can be grouped according to different criteria. In terms of complexity [1], intrinsic methods focus on building inherently interpretable machine learning (ML) models such as decision trees[14]. Post-hoc methods proceed according to a reverse engineering perspective by first building the "black-box" model without considering any complexity constraints, then by adding some sort of interpretable layers such as surrogate models [30] and attributions [36]. In terms of scope, explainability techniques fall into two main categories: Local methods that generate explanations for target predictions using local information around a given instance, and Global methods that explain a prediction based on the entire knowledge learned by the model during training and used during output prediction.

While many works have surveyed the literature on XAI, such as but not limited to [37, 28, 5, 1, 6, 7], there is still no consensus definition or terminology for XAI, due primarily to the subjective nature of the field. The terms "explainability" and "interpretability" are used synonymously [1] in this paper and refer to the same meaning. We define explainability as the field aimed at making predictions derived from ML models more understandable to humans.

Spurred by trends in machine learning, the explainability of deep learning models has in turn taken the lead in XAI. The community has rapidly embarked on the development of explanatory methods for various deep learning-based models, and the number of such approaches continues to grow. According to [28], foundational explainability methods for DNN models can be grouped into three main classes, Visualization, Distillation and Intrinsic. Visualization methods attempt to highlight features of the input that significantly affect the output, such as LRP [3] and Integrated Gradients [36]. Distillation is a class of methods

often known as "white-box" methods, which are supposed to approximate the functioning of the black-box model by a simpler one, such as LIME[30] and Anchors [31]. The last so-called Intrinsic class is similar to the one presented above in this section but has a more general scope. In addition to models that are intrinsically interpretable, there are also models that are capable of generating explanations along with the inference itself, with prototypes [16] being a common method.

It is worth mentioning that some research has used a perturbation paradigm for adversarial attacks and/or training. [19] proposed a score-based attack model for the textual attack task. Their method was based on selecting important words using a self-attention mechanism and generating a correlation degree of words inside the texts, which is to some extent related to the aspect of explainability. In [43], the authors sought to identify the vulnerability of link prediction methods for graph-structured data using a deep architecture-based adversarial attack method. They also investigated other adversarial attack methods such as heuristic and evolutionary perturbation methods. Other methods have also been proposed for various adversarial attack-based tasks such as Graph Neural Networks (GNNs) structure enhancement [41] and time-series prediction [40]. Regarding the image captioning domain, we are only aware of the methods proposed in [45, 51] where the authors focus only on improving the robustness and stability of captioning models, but do not address the explicability issues in the domain.

2.3. Explainability in Image Captioning

The increased awareness of the need to develop more accurate captioning models, mainly based on deep learning, has led to exponential progress in terms of complexity. However, existing work on the explainability for IC remains very limited, with most work adhering to the post-hoc explanation paradigm. The authors in [35] proposed several attribution-based explanation methods, including an adapted version of Layer-wise Relevance Propagation (LRP) [3], as well as Grad-CAM and Guided Grad-CAM [33] that are based on Gradient-

weighted Class Activation Mapping. The new LRP-based method generates pixel-wise visual explanations that highlight supporting and opposing pixels to the prediction of a target word in the output caption, alongside linguistic explanations showing the contribution of each word in the same caption to the prediction of that target word. The idea is that explanations obtained at both visual and linguistic levels of the architecture also called attributions, make the word-region and word-sequence dependencies more explicit.

The authors in [32] employed the LIME [30] surrogate technique to approximate the black-box captioning model by a simpler linear model for each data instance (image in the case of captioning). The intuition is to generate a set of perturbed instances locally around a given image using blur and blackout operations on the original input (pixels), then to evaluate the change in prediction to be included in the cost function of the linear model. Once the model has been trained, it assigns a weight to each input region based on its ability to preserve or not the prediction of a particular word in the caption.

An explainable IC model was also proposed by [10] using an explanation module with a dual role. First, during training, it serves as a loss function that evaluates the correlation between the words generated in the caption and the objects detected in the image and back-propagates the error to enhance the captioning model. In the inference step, the same module generates a weight matrix that assigns an attention coefficient (importance score) to each region in the image relative to every word predicted in the output caption.

Other techniques were proposed for explainability in image classification. The method of [11] generates natural language explanations for classification decisions using class definitions and image descriptions together. Their method was found to produce more relevant explanations that take into account both class-discriminative image features and class definitions. A reinforcement module was also included to condition sentence generation. [9] used the perturbation paradigm in the original input space (the image) to generate meaningful explanations. In their model, they employed an optimization technique that learns perturbation masks that most or least affect the model's output. [46] used mor-

phological fragmentation to divide the input image into multi-scale fragments that are then masked by the perturbation to generate heatmap explanations.

Motivated by the results on visual and linguistic explanations in [35] that consider both components of equal importance, our first work focused on solving a fundamental question regarding the reliability of explanations and their accuracy, which has often been missing in this field. The main idea was to identify and isolate the parts of the architecture that are most involved in the captioning process and to see if decisions are made exclusively on the basis of the inputs or if they are affected by other elements such as the imbalance of influence between the different parts of the captioning architecture. This would then allow the behavior of the internal components to be implicated in the explanation process. Based on the results of this work, we undertook an in-depth study taking into account the specificities of the architecture components. Our target is therefore confined to the visual component for this second part, as it proved to be more decisive than the linguistic part.

It is worth noting that what characterizes the present work compared to the aforementioned works is the use of the latent space to both produce and evaluate explanations. Given the advantages observed in this space, we propose to extend its use in order to provide a comprehensive approach capable of extracting more tangible clues to explain IC decisions, an aspect that has rarely been addressed in the XAI literature. We designed variants of two popular explanation methods, LIME and LRP, both of which adhere to the new latent space paradigm. The choice of these two methods is far from being arbitrary. The objective is to investigate the possible causality between the quality of the explanations and the scope of the explanation methods (local vs global). Moreover, BU-LRP appears to be the first in the field to exploit the BU features, thus focusing on their possible role in providing improved explanations in latent space. As for assessing the quality of explanations, Latent Ablation is also the first method that, in the fashion of the proposed explanation methods, inherits the strengths of latent space. This clearly translates into the manipulation of low-level features considered to be very close to the internal model theory in order to characterize

its functioning, as well as the way the data are encoded. Depending on the granularity of the concepts manipulated in the latent space (i.e. partial as individual features, or full as complete objects), we designed two versions of the LIME-based method and Latent Ablation. The aim is to study the impact of this granularity on the construction and evaluation of explanation methods.

3. Method

This section first presents the common captioning architecture used in this work, and then introduces our approach that identifies the decisiveness and influence of each component of the architecture on captioning decisions. Next, it formally describes the two proposed explanation approaches based on LRP and LIME in the representation space as well as the two evaluation methods, correlation and Latent Ablation. Note that all parameters, such as vector sizes and dimensions, are assigned values in the experiments section.

3.1. Standard captioning architecture

We employ the standard captioning architecture Ada-LSTM from [35] which uses Bottom-Up (BU) image features generated by the object detection module Faster-RCNN. This model is designed upon the common topology of Encoder-Attention-Decoder (Fig. 2), and their work is one of the first to tackle explainability for the IC task, which will provide a reference work for upcoming comparison with our methods. Although more sophisticated architectures exist, such as Transformer-based ones (Sec. 2.1), they might be less interesting for our study, since the high complexity of these architectures relative to their ordinary performances makes them less appropriate for explainability issues, where a trade-off must be preserved [1].

Given an aligned pair (I, S) composed of an image instance I and the corresponding ground-truth captions S, the image encoder extracts a set of BU visual features (latent representations) $\tilde{I}$ whose number is V and dimension is d_v, on top of the set of objects detected from the image by the Object detector.

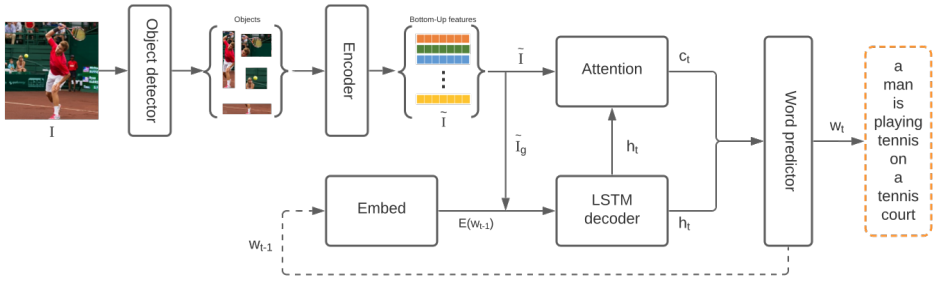

Figure 2: Bottom-Up captioning architecture overview.

Thus, each object (region) is represented by a latent vector v_i (Eq. 1).

$$\tilde{I} = (v_i)_{i=1}^{V} \quad , \quad v_i \in \mathbb{R}^{d_v} \tag{1}$$

The set of visual features is averaged $\tilde{I}_g$, combined to the previous hidden state h_{t-1} and the word embeddings of the previous word of the sequence $E(w_{t-1}) \in \mathbb{R}^{d_w}$, and passed to the LSTM decoder which generates the current hidden state h_t (Eq. 2). d_w and d_h are the dimensions of word embedding and hidden state vectors, respectively.

$$h_t = LSTM(\tilde{I}_g, h_{t-1}, E(w_{t-1})) \quad , \quad h_t \in \mathbb{R}^{d_h} \tag{2}$$

The attention module assigns focus coefficients to every BU feature to generate a context representation c_t whose dimension is d_c (Eq. 3). Then at each time step t, both hidden states and context vectors are used by the language LSTM to predict the next word w_t (Eq. 4) of the output sequence C (Eq. 5), L being the maximum caption length.

$$c_t = ATT(h_t, \tilde{I}) \quad , \quad c_t \in \mathbb{R}^{d_c} \tag{3}$$

$$w_t = WordPredict(h_t, c_t) \tag{4}$$

$$C = (w_t)_{t=1}^{L} \tag{5}$$

3.2. Component influence identification approach

In this section, we introduce our perturbation approach that investigates captioning models to capture the decisiveness and sensitivity of their components, as already presented in our previous work [8]. To do so, we operate on two distinct levels of the architecture (Fig. 3):

- Vision: we inject perturbations on the BU visual features (VF) and context representations (CT) resulting from the vision components of the captioning model, image encoder and attention mechanism, respectively.
- Language: the perturbation concerns the word representations (WE) and hidden states (HT) resulting from the language components, word embedding encoder and the language decoder, respectively.

The intuition behind the specific targeting of these two levels (Vision, Language) by the perturbation is mainly based on their mutual complementarity. The study of the effect of the perturbation on both enables their behavior towards perturbation to be compared, thus covering all parts of the architecture. Indeed, since each level is mainly composed of two components, each of them would need to be carefully examined, intrinsically and compared to the others, to determine its involvement in the captioning process. Furthermore, each of these components embodies an essential subtask to be considered, such as encoding, attention, and decoding.

The perturbation, as already mentioned, is entirely performed in the latent (representation) space of the captioning architecture, and concerns one of the four components at a time. Let $r(\phi)$ denote the perturbation function that performs the element-wise addition of random noise $\eta = (\eta_j)_{j=1}^{d_\eta}$ to a representation vector $\phi \in \mathbb{R}^{d_\phi}$, intuitively $d_\eta = d_\phi$ for each perturbed component. η_j following the Gaussian distribution $\mathcal{N}$ of mean μ and standard deviation σ:

$$\eta_j \sim \mathcal{N}(\mu, \sigma^2) \tag{6}$$

Each η_j is conditioned by Eq. 7 ensuring that the random value falls within the definition domain of the component concerned by the perturbation, bounded

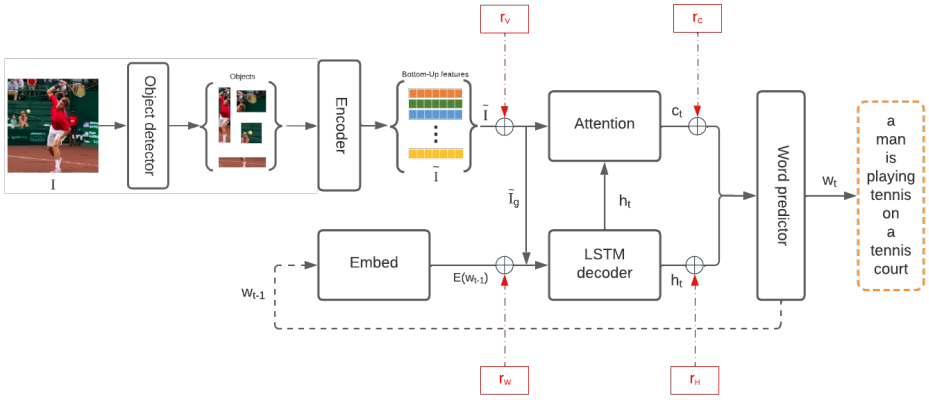

Figure 3: Overview of the perturbation protocol. r_V, r_C, r_W, r_H in red boxes represent perturbations on VF, CT, WE, and HT components respectively. $\oplus$ denotes the direct sum.

by the two values min_G and max_G, respectively the global minimum and maximum.

$$min_G \leq \eta_j \leq max_G \tag{7}$$

3.2.1. Visual level

Visual features: This perturbation concerns the BU visual features at the output of the Faster-RCNN encoder (Eq. 8). To guarantee that one remains in the definition range after the perturbation, a second condition is placed on the perturbed element of the feature vector with respect to the dimension to which it belongs (Eq. 9). v_i represents the i^{th} visual feature as seen in Sec. 3.1, min_j and max_j are the global minimum and maximum for the j^{th} dimension over the entire latent space of the visual feature component.

$$r(v_i) = (v_{ij} + \eta_j)_{j=1}^{d_v} \tag{8}$$

$$min_j \leq v_{ij} + \eta_j \leq max_j \tag{9}$$

Context representations: At the output of the attention module, a context vector c_t indicates the regions that require more attention/focus than others for the generation of the current word of the output sequence. This perturbation

adds random values η to c_t at each time step t (Eq. 10). Unlike the previous component, the values of the context vectors are likely to change at each execution time. The second condition is therefore slightly adjusted for this component to take into account the global minimum and maximum instead of those per dimension (Eq. 11).

$$r(c_t) = (c_{tj} + \eta_j)_{j=1}^{d_c} \tag{10}$$

$$min_G \leq c_{tj} + \eta_j \leq max_G \tag{11}$$

3.2.2. Language level

Word embeddings: In the same way, we define the perturbation of the word embeddings at the output of the embedding module. As shown in Eq. 12, a random perturbation η is added at each time step t to the embeddings of the predecessor w_{t-1} of the word to be generated. The same condition as for the context representation is applicable.

$$r(E(w_{t-1})) = (E(w_{(t-1)j}) + \eta_j)_{j=1}^{d_w} \tag{12}$$

Hidden states: This perturbation involves the hidden representations generated by the LSTM decoder, which encapsulates the information from the previous predicted sequence to word $wt - 1$ (Eq. 13). The same condition as for context representation must be complied with.

$$r(h_t) = (h_{tj} + \eta_j)_{j=1}^{d_h} \tag{13}$$

3.3. Explanation method based on LRP attributions

We showed in our previous work [8] that the visual part of IC models constitutes a key element for performing subsequent explanations on these models. In this section, we further explore the visual modality to extract more explicit clues about the decisions made by the captioning model. To do so, we continue the work of [35] by adapting the LRP explanation method to IC models that use BU visual features extracted by Faster-RCNN rather than the classical global CNN feature. Note that what differentiates the BU features from the CNN feature is that the former depend on local pixels in the input image, so that each

feature encodes a specific region, while the latter depends on all pixels and thus encodes the entire image in a global feature. The idea is to identify fine-grained elements that could explain IC decisions based on the objects that compose the image.

The concept of LRP is similar to that of backpropagation. Given a neural network, LRP back-propagates the final prediction (output) along the network by recursively assigning a relevance score to each neuron in the network, until the input is reached. Specifically, the relevance of each neuron to the final prediction is decomposed to each of the neurons in the previous layer according to several attribution rules defined by [3].

Consider a neural network composed of a set of neural layers connected in a layer-wise manner, where a neuron x_j of the layer (l) is defined as a linear transformation of all neurons of the previous layer $(l\text{-}1)$ followed by an activation $g(\cdot)$ (Eq. 14). x_i is an input neuron, y_j the linear output and x_j the activation output.

$$x_j = g(y_j) \;\; ; \;\; y_j = \sum_i x_i w_{ij} + b_j \tag{14}$$

$$R_{i \leftarrow j}^{(l\text{-}1,l)} = R_j^{(l)} \;\cdot\; \frac{x_i w_{ij}}{y_j \;+\; \epsilon} \tag{15}$$

$$R_i^{(l\text{-}1)} = \sum_j R_{i \leftarrow j} \tag{16}$$

Given the known relevance score $R_j^{(l)}$ of the neuron x_j in the layer (l), a decomposition rule called $\epsilon - rule$ (Eq. 15) assigns a contribution score $R_{i \leftarrow j}^{(l\text{-}1,l)}$ to each input neuron x_i in the previous layer $(l$ - $1)$. Intuitively, this can be thought of as the relative contribution of the neuron x_i to all neurons in the same layer, for the computation of the neuron x_j. The parameter ϵ works as a stabilizer to avoid unbounded values of the relevances $R_{i \leftarrow j}$ when y_j takes small values. Finally, the global relevance score of the neuron x_i is obtained by summing all the incoming contributions from the layer (l) (Eq. 16).

In the Ada-LSTM captioning model from [35] that we use, which is mainly composed of a Faster-RCNN encoder, an adaptive attention module and an

LSTM decoder, we adapted the LRP explanation method as follows to match the BU captioning architecture. The authors in [35] stated that, as far as the entire captioning architecture is concerned, LRP rules follow the same topological flow as backpropagation (component by component). In this regard, we follow the same logic and initialize the relevance scores of the words in the output caption with the logits of the last layer of the word predictor sub-module. We back-propagate each word's relevance score $R(w_t)$ along the architecture through the Word predictor, LSTM decoder and Attention. We stop at the output of the encoder as shown in Fig. 4 to obtain the relevances of all BU visual features previously used for prediction in the forward pass, whose number is V.

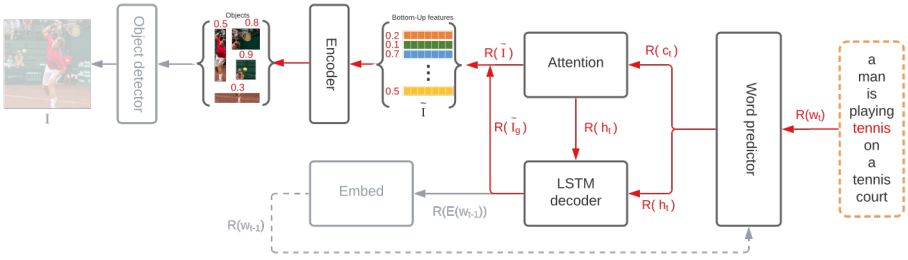

Figure 4: LRP backpropagation flow through the captioning architecture with BU features. The feature representing a tennis ball got the highest importance score to the prediction of the word tennis in the caption.

At the output of each component (in the backpropagation direction), we obtain the relevance scores of the corresponding data flow. The notations $R(c_t)$, $R(h_t)$, $R(\tilde{I}_g)$, $R(\tilde{I})$ refer to the relevance scores of the context vectors, hidden states, global image feature and BU image features respectively. We are only interested in exploring the relevance of the BU features for final word prediction, as the linguistic modality has already been shown (in Sec. 3.2) to be of lesser importance to the explanation process.

The vector of relevance scores $R(v_i)$ for each BU feature v_i obviously takes the same shape as the feature vector, i.e. $R(v_i) \in \mathbb{R}^{d_v}$. The global relevance score for one BU feature is then the summation of all the elements $R(v_{ij})$ of

the relevance vector. We obtain a coefficient vector $\hat{\alpha} \in \mathbb{R}^V$ as a final explanation output (Eq. 17), containing the importance of all the BU features to the prediction of a given word in the caption.

$$\hat{\alpha} = R(\tilde{I}) = \{R(v_i)\}_{i=1}^{V} \ , \ \ R(v_i) = \sum_{j=1}^{d_v} R(v_{ij}) \tag{17}$$

3.4. Explanation method with LIME attributions

In this part, we introduce our method BU-LIME by adapting LIME[30] to the IC task that uses BU visual features. LIME is a perturbative method that can be used to explain any black-box model. It works by perturbing the original input to generate neighbor instances, that are used to predict new outputs and evaluate the change with respect to the original output. This evaluation is achieved by training a linear model whose inputs and outputs are the perturbed instances and their corresponding outputs. The coefficients of the linear model provide local explanations in the form of importance attributions to every input feature for predicting a particular output.

To avoid any confusion, we call *Intrinsic perturbations*, the perturbations that are part of the LIME method itself, unlike the Gaussian perturbations previously introduced in Sec. 3.2. To build BU-LIME, we follow the same reasoning as in the original LIME but with a different perturbation technique. Indeed, we opt for gradual intrinsic Gaussian perturbations in the latent space (the visual features) that have been shown to be more relevant than altering the original image by means of ablations, which could lead to truncation or loss of information due to blurring and blackening operations as in [32].

First, we generate a set of P neighborhood instances $\Gamma = \{\tilde{I}^{(p)}\}_{p=1}^{P}$ around a given image I. Each perturbed instance is obtained by adding random perturbations to a subset of its visual features. There are various ways to choose this subset that we will introduce in the experimental part, but in general randomness while choosing the features to be perturbed and consistency while fixing their number must hold. A binary vector $X^{(p)} \in \{0,1\}^V$ whose elements are set to '1' for the perturbed features, '0' elsewhere, is associated to every perturbed

instance $\tilde{I}^{(p)}$ via an indexation module, V being the number of BU-features. Each perturbed instance is then fed into the captioning model. The set of all the binary vectors $X = \{X^{(p)}\}_{p=1}^{P}$ can be denoted as the binary matrix $X \in \{0,1\}^{P \times V}$. The generated captions as such are not coherent and thus not useful since the corresponding inputs have been subject to perturbation. We are instead interested by the generated weights (logits from the last layer of the LSTM) of all the words belonging to the vocabulary while predicting the output caption. Considering that a weight for each word is obtained at each decoding step, we take the maximum over all the decoding steps as the final weight, e.g. for the first perturbed instance, if we consider that the maximum number of decoding steps is 5 and we get the following weights for the word 'man': 0.5, 0.3, 0.33, 0.4, 0.7; then the final weight for that word is: 0.7; this process is performed for all the words of the vocabulary, for each perturbed instance. At the end of this stage, we get a weight matrix $\Delta = (\delta_{pq}) \in \mathbb{R}^{P \times Q}$, Q being the size of the vocabulary.

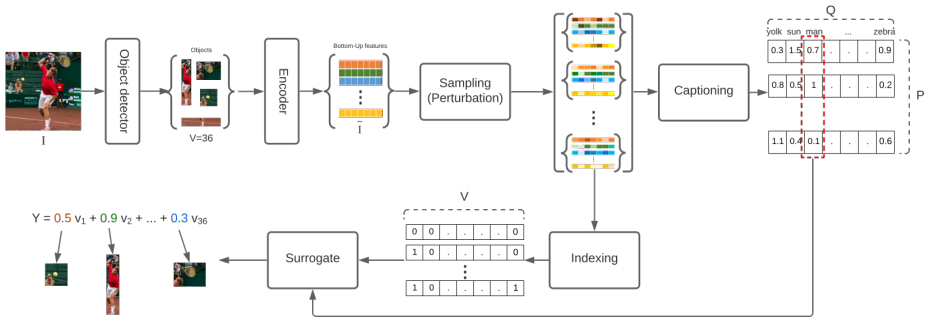

Figure 5: BU-LIME method for BU based IC models. The coefficients of the linear model reflect the importance of each visual feature to the prediction of the word 'man'. The highest coefficient corresponds to the feature that represents a man.

To get the explanations for a given word w_q belonging to the vocabulary, we build a linear regression model (Eq. 18) using the paired data (Binary matrix X, Weight vector y) as the training set, $y = \delta_{.q} \in \mathbb{R}^P$ being the q^{th} column of

Δ. A single training instance is thus represented as a pair (X_p, y_p). The cost function and the objective function of the regression model are given by Eq. 19 and Eq. 20 respectively. $\beta \in \mathbb{R}^V$ is the vector of coefficients to be estimated and $\hat{\beta}$ is the estimator, $\gamma \in \mathbb{R}^V$ is a noise vector. The entire approach is illustrated in Fig. 5 and synthesized in Algorithm 1.

$$Y = X \cdot \beta + \gamma \tag{18}$$

$$cost = \frac{1}{P}\|Y - X\beta\|_2^2 + \lambda\|\beta\|_2^2 = \frac{1}{P}\sum_{p=1}^{P}(\ y_p - \sum_{v=1}^{V}(X_{pv} * \beta_v)\ )^2 + \lambda\sum_{v=1}^{V}{\beta_v}^2 \tag{19}$$

$$\hat{\beta} \in argmin(cost) \tag{20}$$

Algorithm 1 LIME for BU captioning model to explain w_q. Notations: $\hat{\beta}$ is the explanation vector (Eq. 20), q is the index of the word to be explained, Γ is the set of all perturbed instances, X is the set of all perturbation indexes, y is the vector of the predicted captioning weight. *perturb*, *capt* and *linear* are the perturbation, captioning and linear regression functions, respectively.

Inputs: $\tilde{I}, q$

Outputs: $\hat{\beta}$

$\Gamma \leftarrow \{\varnothing\}$

$X \leftarrow \{\varnothing\}$

for $p \in \{1, ..., P\}$ **do**

  $\tilde{I}^{\acute{}(p)}$, $X^{(p)} \leftarrow perturb(\tilde{I}^{(p)})$

  $\Gamma \leftarrow \Gamma \cup \{\tilde{I}^{\acute{}(p)}\}$

  $X \leftarrow X \cup \{X^{(p)}\}$

  $\delta_{p.} \leftarrow capt(\tilde{I}^{\acute{}(p)})$

end for

$y \leftarrow \delta_{.q}$

$\hat{\beta} \leftarrow linear(X, y)$

return $\hat{\beta}$

Knowing that regions/objects of an image are encoded by several visual features (sometimes implying redundancy or even ambiguity), we propose to study the effect of manipulating these features, through a second version of the LIME-based explanation method that involves objects rather than individual visual features. The difference lies in the way the intrinsic perturbation of LIME is done. The new version called BU-LIME-N-OBJ involves performing a full perturbation for up to N objects. The number of perturbed instances per image is then given by $P = \sum_{i=0}^{N} C_n^i$, n being the number of objects detected in the image. Obviously, we retain only the images with $n \geq N$. We experiment with two values of N (5 and 8). The reason we do not use lower values for N is that the number of instances would be insufficient to train the LIME linear model (example: for N=2 and n=7, p=29). Note that we empirically found that the maximum number of unique objects in an image is 15 (not counting overlapping/redundant regions), for which reason we perturb at most half the objects to avoid excessive perturbation. A larger number of perturbed objects has been shown experimentally to penalize the learning of the linear model. We refer to the example in Fig. 6 to better understand this notion of redundant visual features.

3.5. Evaluation of explanations quality

Existing work on IC explainability assessment is primarily based on a qualitative [44, 13, 10] and quantitative [35] evaluation of the fidelity property of the attributions (eg. Saliency maps) generated by the perceptual explanation techniques, which include either a visual assessment of the correctness of the explanation related to the explained item, or the measurement of the discrepancy between a new prediction, while removing the explanation from the input, and the original prediction. In this section, we propose two new evaluation methods, the first of which is based on measuring the correlation between the explanations and the object detection scores. The second is based on the Latent Ablation principle that we briefly mentioned in Sec. 1. Both methods are fully described in what follows.

3.5.1. Correlation measure

This metric measures the correlation between the explanations and the classification scores of the different regions (objects) jointly generated by the object detector during image encoding. However, this evaluation is not performed as an absolute comparison between the explanations generated by the attribution methods and the object detector probabilities (Faster-RCNN). Instead, we have developed an evaluation approach that projects the compared items into a unified ranking space.

Consider the candidate (predicted) caption $\hat{C}$ for a given image I. Let $O_{\hat{C}}$ be the list of words representing the objects that appear in $\hat{C}$. Let O_V be the list of labels representing the visual objects detected by the Faster-RCNN. The evaluation will only take into account the explanations of the words of $O_{\hat{C}}$ which appear jointly in O_V. The idea is to compute a distance that reflects the difference between the importance of a given region v_i to the prediction of its label $l \in \{O_{\hat{C}} \cap O_V\}$ in the output caption, and the class (label) probability predicted by the object detector. For this purpose, we used the explanation vectors $\hat{\alpha}$ and $\hat{\beta}$ resulting from BU-LRP and BU-LIME methods respectively applied to word l, which contain the attribution (importance) scores of the BU features. The elements of each vector are sorted and indexed in descending order (*sort_index* function), giving $E_{\hat{\alpha}}$ and $E_{\hat{\beta}}$ respectively (Eq. 21). Thus, the indices of the features (regions) considered as the most relevant for the prediction of the word l will be placed in the first positions and inversely for those of lesser importance. The evaluation consists in measuring the distance that separates the position of the region v_i in both $E_{\hat{\alpha}}$ and $E_{\hat{\beta}}$ (*position* function), from the one given by its ranking at the output of the Faster-RCNN (i.e. the index i). Note that the regions at the output of the Faster-RCNN are ranked in descending order according to their class probabilities, i.e. from most to least probable.

$$E_{\hat{\alpha}} = sort\_index(\hat{\alpha}) \;\; ; \;\; E_{\hat{\beta}} = sort\_index(\hat{\beta}) \tag{21}$$

$$d_i^{\hat{\alpha}} = max(\, 0 \,,\, position(E_{\hat{\alpha}}, v_i) - i \,) \tag{22}$$

$$s_i^{\hat{\alpha}} = \frac{1}{1 \ + \ d_i^{\hat{\alpha}}} \in [0, 1] \tag{23}$$

Nevertheless, if the position of v_i in the sorted vectors $E_{\hat{\alpha}}$ and $E_{\hat{\beta}}$ is better than or equal to the one given by the object detector (in terms of importance), the distance is kept to the minimum value '0' (Eq. 22). This means that the explanation produced by the concerned method has a correlation factor with as much preciseness as the classification score between the predicted label and the corresponding BU feature. The obtained distance is finally transformed into a similarity score between the explanation and the object detection score, expressed by Eq. 23.

3.5.2. Latent Ablation measure

In addition to the correlation metric defined above, we also propose Latent Ablation. The original concept of ablation consists in masking specific parts (regions) of the image considered as the most relevant for the prediction of a given word in the output caption. The idea is to eliminate their contribution to the prediction and evaluate the change in the quality of the output caption. Herein, we propose a similar concept called *Latent Ablation*, where we operate on the latent space (i.e. features) rather than the original input space[35] (i.e. images).

Analogous to what we proposed previously in Sec. 3.4 regarding the two versions of the LIME-based explanation approach (visual features VS object), we propose here to handle two versions of Latent Ablation, by masking either the top k visual features or the top k entire objects, still in the latent space. This makes it possible to pinpoint the role of objects in evaluating the explanations provided by the proposed methods.

Masking the top k visual features amounts to ablating those considered most relevant for the final prediction, with the rest of the features left unchanged. We set up various ablation magnitudes, with the first operating by masking the visual features by means of Gaussian perturbations with some standard deviation. The other magnitudes saturate the visual features with either the

minimum or the maximum values of the VF component (see Sec. 4.2.1). Afterward, we re-generate the image caption and check for the presence (or not) of the word concerned by the explanation in the new caption. We consider all the object words for the evaluation and ignore stop words and predicates. We report the final results as a percentage of missing words that reflects the fidelity of the explanation items to the corresponding predictions.

The second version of ablation considers masking top k objects rather than individual features. An object is represented by a set of visual features and a visual feature corresponds to only one object in the image. Multiple visual features may be generated for a given object in the image, as Faster-RCNN may detect partial areas of it, obviously with lower probabilities for their labels than for the entire object. Figure 6 shows the detection results on an image representing a man playing tennis, where the main object "man" is delimited by multiple bounding boxes (man 83%, man 80%, man 35%...), and thus represented by multiple visual features in the latent space, most of which are fragments of the whole object "man" that has received the highest probability.

The top k objects are chosen according to the rank of their first occurrence in terms of visual features in the explanation vector (composed of indices representing the 36 extracted features, ranked by importance). This first occurrence thus reflects the most likely feature among all the extracted features corresponding to this same object.

4. Experiments and results

In this section, we first introduce the dataset and the captioning model used for this study. We then provide the experimental settings, results and discussion for the component influence identification part. Finally, we present and discuss the results of the attribution-based methods, BU-LRP and BU-LIME.

4.1. Dataset and captioning model building

For all the experiments, we employed the standard captioning architecture based on BU features from [35]. This architecture is composed of a Faster-RCNN

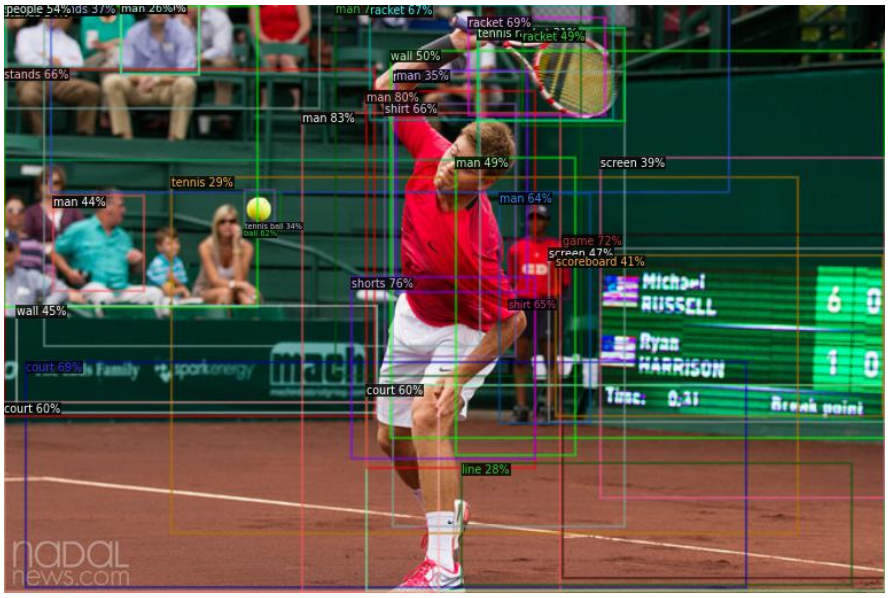

Figure 6: Illustrative example of the object detection results using Faster-RCNN.

encoder, an adaptive attention module, an LSTM decoder and an LSTM word predictor. We retrained the model on the train partition of the MSCOCO2017[18] dataset (110000 instances) and the train partition of the Flickr30k[48] dataset (29000), and automatically stopped the training once we obtained the best CIDEr score (at the 21^{th} and 8^{th} epochs, respectively). The dataset is aligned on (image-captions) pairs with $k = 5$ ground-truth captions per image. A vocabulary of words whose size $Q = 11026$ is built from the captions of the training sets.

We used the Detectron2 [42] implementation of Faster-RCNN to extract $V = 36$ BU visual features, whose dimension is $d_v = 1024$, representing the V visual regions detected in each image. The dimensions of context vectors, word embeddings and hidden state vectors were set to $d_c = 512, d_w = 512, d_h = 1024$. The maximum length of caption sequences was $T = 20$. The results of the different experiments are reported on the test partition of MSCOCO2017 (5000 instances) and Flickr30k (1000 instances).

4.2. Component influence analysis

4.2.1. Perturbation settings

We experimented with Gaussian perturbations with mean $\mu = 0$ and different values of standard deviation σ. We empirically computed the upper and lower bounds of each component involved in the perturbation. These interval bounds allow us to determine the values of σ based on the definition domain of the perturbed component's representation space as shown in Tab. 1. Intuitively, the σ controls the magnitude of the perturbations to be added to the values of each component. We initialize sigma with the highest value it can take (that gives the maximum perturbation as described in Tab. 1), and at each step, we halve this value and perform the perturbation until we reach stationarity in the quality of the captions. We retain only the sigma values that are the most representative of the results and repeat the process for all components. The following sigma values were reported at the end of the experiment: (1.5, 0.75, 0.375, 0.1875) for VF, CT, and WE ; (0.375, 0.1875, 0.0938, 0.0469) for HT.

Table 1: Global interval bounds and standard deviation values according to the perturbed component.

	Dataset	VF	WE	CT	HT
$[min_G, max_G]$	MSCOCO2017	[0, 16.5]	[-5, 5]	[-1, 9.5]	[-1, 1]
	Flickr30k	[0, 16.5]	[-5, 5]	[-1, 14.6]	[-1, 1]
$\sigma \leq$		1.5	1.5	1.5	0.375

Figure 7 illustrates qualitative results obtained by performing perturbation on instances from the MSCOCO2017 test set. As can be clearly seen, the captions from the VF perturbation show the most divergence from the reference and do not reflect the image content. The CT perturbation is able to generate some basic concepts such as the color of the mug "white" (left example) and "room" (right example) but still fails to produce a coherent/complete caption. We finally managed to generate almost all the necessary elements with both WE and HT perturbations.

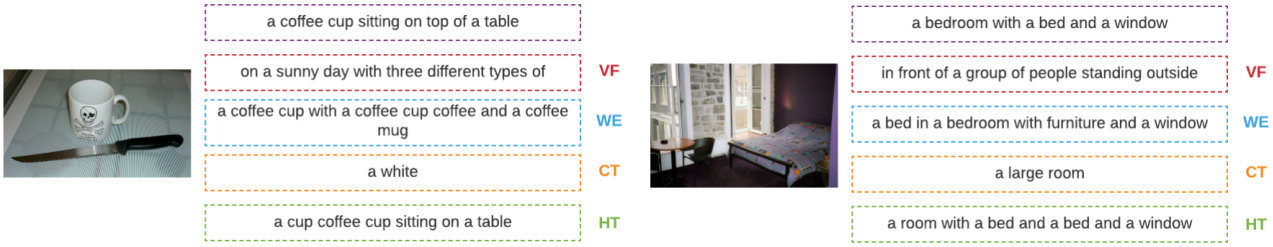

Figure 7: Qualitative examples of the perturbation results on two MSCOCO2017 test instances using the maximum perturbation magnitude. Captions in purple boxes correspond to the system references. Each of the captions in the bottom boxes corresponds to the perturbation indicated alongside.

4.2.2. Evaluation of caption quality

We used the most common evaluation metrics for the IC task to assess the quality of captions, specifically BLEU [27], METEOR [4], CIDEr [39], SPICE [2] and ROUGE [17]. We also used MSICE, a new evaluation metric that we proposed in our previous work, which takes into account two important linguistic aspects: morphology and semantics, contrary to some existing metrics based only on n-gram overlapping such as BLEU, ROUGE-L, and CIDEr. We refer to [8] for a detailed description.

We repeated our experiment 30 times on both MSCOCO2017 and Flickr30k test sets to control the randomness effect. The quality of the captions is subject to a double evaluation. *Machine-Human (M-H)* evaluation allows a comparison of the system-generated captions after perturbation with the ground-truth human captions (references), whereas a *Machine-Machine (M-M)* evaluation compares the captions generated by the model after perturbation with the ones before perturbation. With M-M evaluation, results can be compared at the system level without involving the human factor to measure the gap between machine and human conclusions. Complementary to the ones presented in our previous paper [8], the detailed results are only reported on the Flickr30k test

set in Tabs. 2-5 (M-H evaluation). However, for comparison purposes, we included curves that synthesize the results obtained on the test partitions of both Flickr30k and MSCOCO2017 in Fig. 8 and Fig. 9 respectively (M-H evaluation). Figure 10 finally shows the results of an M-M evaluation on the Flickr30k test set.

Table 2: M-H evaluation of visual features perturbation on Flickr30k test set.

σ	BLEU-4	CIDEr	SPICE	ROUGE	METEOR	MSICE
0.1875	0.2329	0.4763	0.1467	0.4576	0.2046	0.4971
	0.0013	0.0028	0.0004	0.0008	0.0005	0.0018
0.3750	0.204	0.419	0.1353	0.4405	0.1935	0.4771
	0.0015	0.0026	0.0006	0.0009	0.0008	0.002
0.7500	0.1596	0.307	0.1118	0.4096	0.1763	0.4229
	0.0017	0.0035	0.0009	0.001	0.0006	0.0026
1.5000	0.1154	0.159	0.0858	0.3728	0.1591	0.3514
	0.0009	0.002	0.0008	0.0007	0.0007	0.0032

Table 3: M-H evaluation of word embedding perturbation on Flickr30k test set.

σ	BLEU-4	CIDEr	SPICE	ROUGE	METEOR	MSICE
0.1875	0.2535	0.5245	0.1527	0.4709	0.2097	0.5093
	0.0025	0.0064	0.0012	0.0016	0.0011	0.0024
0.3750	0.2520	0.5190	0.1520	0.4694	0.2089	0.5088
	0.004	0.0067	0.0014	0.0024	0.0012	0.0035
0.7500	0.2446	0.5001	0.1484	0.4638	0.2054	0.5010
	0.0043	0.0088	0.0018	0.0025	0.0017	0.0039
1.5000	0.1767	0.3622	0.1193	0.4154	0.1734	0.4347
	0.0044	0.0096	0.0017	0.0032	0.0017	0.0054

4.2.3. Discussion on component influence

Observing the scores in Tabs. 2-5 (M-H evaluation), one can initially notice a significant decrease in the quality of the captions for the visual feature (VF) and context representation (CT) components, in contrast to hidden states (HT) and word embeddings (WE), as the perturbation amplitude increases.

Table 4: M-H evaluation of context vectors perturbation on Flickr30k test set.

σ	BLEU-4	CIDEr	SPICE	ROUGE	METEOR	MSICE
0.1875	0.2456	0.5117	0.1509	0.4660	0.2076	0.5059
	0.0026	0.0065	0.0013	0.0029	0.0015	0.0040
0.3750	0.2236	0.4705	0.1435	0.4505	0.2010	0.4931
	0.0042	0.0073	0.0014	0.0030	0.0016	0.0049
0.7500	0.1485	0.3244	0.1192	0.3972	0.1767	0.4416
	0.0037	0.0096	0.0022	0.0031	0.0017	0.0054
1.5000	0.0247	0.0583	0.0352	0.2347	0.0951	0.1730
	0.0023	0.0026	0.0012	0.0020	0.0012	0.0054

Table 5: M-H evaluation of hidden states perturbation on Flickr30k test set.

σ	BLEU-4	CIDEr	SPICE	ROUGE	METEOR	MSICE
0.0469	0.2553	0.5272	0.1534	0.4706	0.21	0.509
	0.0023	0.0047	0.0009	0.0014	0.0008	0.0034
0.0938	0.2524	0.5246	0.1530	0.4704	0.2100	0.5100
	0.0030	0.0072	0.0012	0.0020	0.0012	0.0029
0.1875	0.2496	0.5165	0.1521	0.4683	0.2090	0.5087
	0.0035	0.0072	0.0014	0.0024	0.0010	0.0032
0.3750	0.2358	0.4834	0.1476	0.4592	0.2036	0.4982
	0.0038	0.0076	0.0014	0.0027	0.0015	0.0046

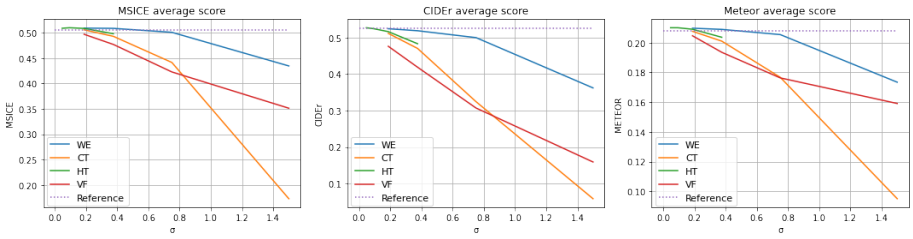

Figure 8: The average scores of MSICE, CIDEr and METEOR with an *M-H* evaluation on the Flickr30k test set. Higher scores mean lower sensitivity hence lower influence and vice-versa. Curves in purple represent the reference scores (i.e. without perturbation).

The standard deviations between the results of the 30 iterations on the different components are extremely low, indicating that each iteration behaved similarly.

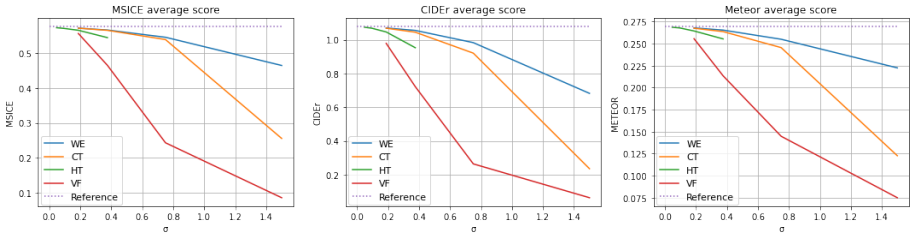

Figure 9: The average scores of MSICE, CIDEr and METEOR with an *M-H* evaluation on the MSCOCO2017 test set.

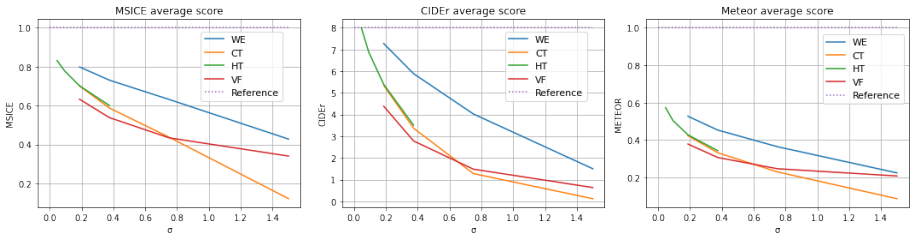

Figure 10: The average scores of MSICE, CIDEr and METEOR with an *M-M* evaluation on the Flickr30k test set.

Figure 8 illustrates the curves drawn for all components based on the three metrics MSICE, CIDEr, and METEOR. Globally, similar behavior is observed on the Flickr30k dataset compared to the MSCOCO2017 dataset. This confirms, with regard to the decisiveness and the influence of the architecture components, the findings drawn from the influence study in our previous paper [8], namely that the VF and CT components which constitute the visual part of the IC model are more decisive than the language components, HT and WE. An important fact derived from the aforementioned results is that results of the explanation methods that consider the visual and language parts of IC models to be of equal contribution while generating the output captions need to be refined. For example, the authors in [35] generate explanations for the prediction of a given word in the output caption at both visual and linguistic levels. However,

our study shows that the language part is not of much importance, in terms of explanations, to the prediction. Rather, it is the visual part that gives clues of a finer degree that highlight the contribution of visual information (regions in the image, objects...).

A slight difference could be observed in the shape of the orange curve representing the contextual component in Fig. 8 (Flickr30k dataset) compared to that in Fig. 9 (MSCOCO2017). This can be traced back to the difference in the context nature of the images that each dataset represents, e.g. Flickr30k mainly focuses on people and animals. Thus, it is expected to be more sensitive to context perturbation than MSCOCO2017 which contains more diverse concepts (objects). The M-M evaluation in Fig. 10 shows similar results but, unlike the M-H evaluation, we observe a concave shape for the different curves, which could be produced by the nature of the comparison and the references that were used (machine references instead of human ones).

By virtue of these results, one would expect the visual component to represent a pivotal element in the next stages of explainability, which is the main focus of the next section.

4.3. Attribution-based explanations

In this section, we evaluate and compare the explanations provided by the two attribution methods for captioning models, BU-LRP and BU-LIME, using the two evaluation methods in Sec. 3.5, the correlation of the explanations to object detection scores and Latent Ablation, that we specifically designed for IC explainability. Both BU-LRP and BU-LIME operate on the representation space and rely on the visual component to derive explanation elements. However, they differ in the nature of their scope, with a global scope for BU-LRP versus local explanations for BU-LIME. So, it is worth comparing the precision of their results with respect to this criterion.

Figure 12 shows an example of a good explanation with BU-LRP for the word "giraffe" in the image caption of Fig. 11. We recall that these explanations reflect the back-propagated contribution through the captioning architecture starting

with the end outcome, i.e. probability of the predicted word in the output caption, up to obtaining the importance of each element of the BU visual feature vector (Fig. 12(a)), and then the overall importance of the visual feature when summed/averaged (at the top of the heatmap). We get 36 importance vectors for each word in the caption, each corresponding to a visual feature (Fig. 12(b)).

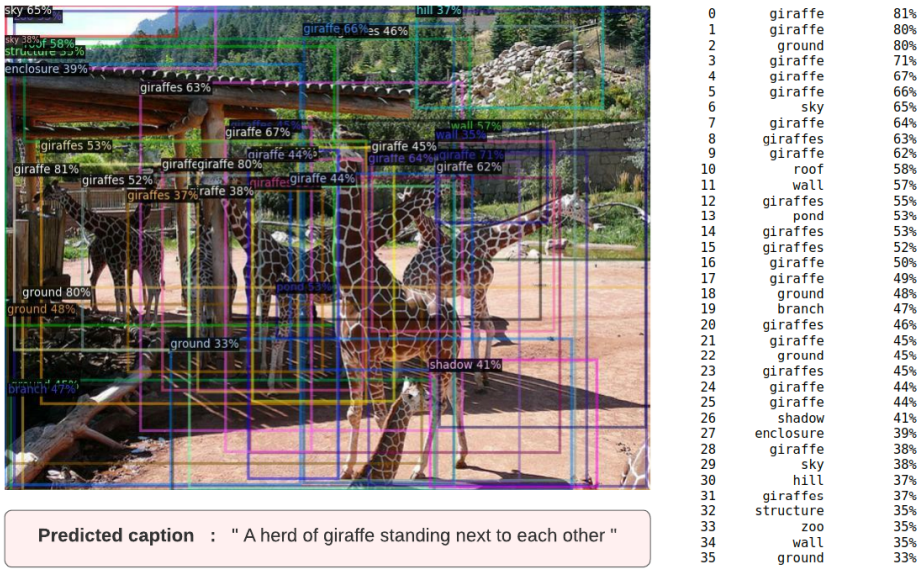

Figure 11: Example of object detection on an image representing a herd of giraffes.

Note that for simplicity and visualization purposes, the importance vector (of size 2048) is displayed as a (64*32) matrix as shown in Fig. 12(a). As can be observed, the explanation values of the same vector are homogeneous, as are the colors of the heatmap (matrix) for the same feature, except for a few outliers, which may suggest an over/under contribution of some dimensions in a given feature. In Fig. 12(b), we can see that some visual features have a higher impact on the prediction of the word “giraffe”, which is depicted by a heatmap with warmer color (9^{th}, 10^{th}, 18^{th}, 21^{st}, 27^{th} and 28^{th} features which respectively

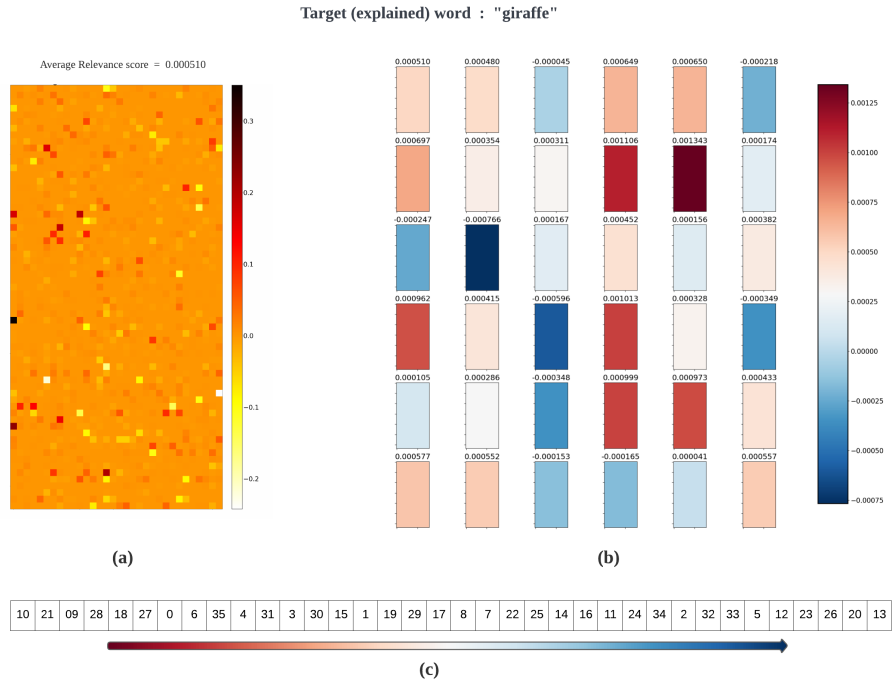

Figure 12: Visualizations of explanations obtained for the word "giraffe" in the caption of the example in Fig. 11. (a) Importance distribution of a single visual feature (v_0). (b) Importance of all visual features (v_0 to v_{35} respectively from top left to bottom right). (c) Explanation vector resulting from the BU-LRP. Warm colors indicate higher importance and vice versa.

correspond to the objects/regions with the labels: giraffe, roof, ground, giraffe, enclosure, giraffe). The visual features ranked by BU-LRP relevance score from most important to least important are presented in Fig. 12(c) for further clarity.

4.3.1. The correlation of explanations to object detection scores

Table 6 reports the evaluation results on the MSCOCO2017 test set, expressed in terms of the total average distance $\overline{d}$ that separates an explanation's position from the rank given by the Faster-RCNN and the average correlation score $\overline{s}$, as described in Sec. 3.5.1. The *common_words* and *non_common_words* correspond respectively to the number of times the word to be explained appears in the list of detected objects (thus taken into account in the calculation

of the scores) or not (word to be ignored). We experimented with two BU-LIME models which differ in the way we generate the perturbed instances. For the first model BU-LIME-1-2, we performed a comprehensive perturbation (all possible combinations) for up to two visual features at a time and obtained a total number of instances $P = 676$ per image, one of which is without any perturbation. BU-LIME-5 is built based on a random selection of five features at a time. Since the number of possible combinations is very large (C_{36}^{5}) and the execution time is proportional to the number of features, we had to limit the number of perturbed instances to $P = 50$ per image. The results presented in Tab. 6 will be discussed later in Sec. 4.3.3.

Table 6: Global correlation scores for the explanations provided by BU-LRP and BU-LIME.

	common_words	*non_common_words*	$\overline{d}$	$\overline{s}$
BU-LRP	8256	6422	10.2633	0.3754
BU-LIME-1-2			13.9510	0.3492
BU-LIME-5			13.7025	0.3418

4.3.2. Latent Ablation

The Latent Ablation experiments were conducted using the same captioning model (Ada-LSTM) that we used overall in this study. We report the results on the MSCOCO2017 and Flickr30k test sets. Table 7 shows the results of the **visual features ablation** for all explanation methods. The normal ablation magnitude corresponds to perturbing the visual features by adding Gaussian values with $std = 1.5$ (the maximum value that std can reach for the VF component), the min and max magnitudes correspond to substituting the feature values by the minimum and maximum values of the vector dimension concerned, respectively. We also include a random ablation baseline (random explanations) for comparison purposes.

Table 8 shows the ablation results for **object ablation** reported in terms of the percentage of missing words and two additional metrics: the average probability drop which expresses the amount of drop in the probability of prediction of

Table 7: Explanation coherence scores for features ablation experiments on BU-LRP and BU-LIME explanations, expressed in terms of percentage of missing words. k is the number of visual features masked at a time.

k	Explanation method	Ablation magnitude		
		normal	min	max
1	Random	0.1159	0.2823	0.2735
	BU-LRP	**0.1211**	**0.2840**	**0.2756**
	BU-LIME-1-2	0.1057	0.2797	0.2619
	BU-LIME-5	0.1089	0.2807	0.2648
3	Random	0.1403	0.5116	0.6094
	BU-LRP	**0.1445**	0.5170	**0.6125**
	BU-LIME-1-2	0.1160	**0.5189**	0.6015
	BU-LIME-5	0.1224	0.5154	0.5997
6	Random	0.1779	0.6881	0.8452
	BU-LRP	**0.1948**	0.6892	**0.8456**
	BU-LIME-1-2	0.1616	**0.6944**	0.8364
	BU-LIME-5	0.1559	0.6847	0.8340
9	Random	0.2338	0.7700	**0.9019**
	BU-LRP	**0.2418**	**0.7744**	0.9012
	BU-LIME-1-2	0.2029	0.7725	0.8945
	BU-LIME-5	0.1985	0.7703	0.8910

a given word in the new caption (after ablation) with respect to the one before ablation, and the probability drop frequency which calculates the number of times a decrease in this probability occurs. Note that for brevity, only ablation with the min value of the visual component has been reported in Tab. 8.

4.3.3. Discussion on attribution explanations

Based on the results of Tab. 6 in Sec. 4.3.1, the correlation scores of the different explanation models do not look promising at first glance. The average distance $\overline{d}$ separating the position (importance) of the explanations from the predictions of Faster-RCNN is approximately one-third of the total number of positions (36), and so is the correlation score $\overline{s}$. This means that the explana-

Table 8: Explanation coherence scores for object ablation experiments on BU-LRP and BU-LIME explanations.

k	explanation method	Avg Prob drop	Prob drop Freq	% of missing words
	Random	1.3521	0.9028	0.4460
1	BU-LRP	2.1259	**0.9311**	0.5931
	BU-LIME-1-2	**2.1548**	0.9242	0.5922
	BU-LIME-5	2.1322	0.9224	**0.5960**
	BU-LIME-5-OBJ	1.3781	0.9011	0.4568
	BU-LIME-8-OBJ	1.1813	0.8908	0.4191
	Random	2.8782	0.9507	0.7309
3	BU-LRP	3.4346	0.9595	0.8033
	BU-LIME-1-2	**3.4776**	**0.9600**	0.8068
	BU-LIME-5	3.4396	0.9589	**0.8091**
	BU-LIME-5-OBJ	2.4333	0.9417	0.6598
	BU-LIME-8-OBJ	2.1437	0.9374	0.6284
	Random	3.4810	0.9332	0.8061
5	BU-LRP	3.8915	0.9623	0.8617
	BU-LIME-1-2	**3.9280**	**0.9640**	**0.8677**
	BU-LIME-5	3.9019	0.963 3	0.8654
	BU-LIME-5-OBJ	2.8814	0.9486	0.7414
	BU-LIME-8-OBJ	2.5787	0.9458	0.7097

tions provided by the explanation method showing the most important features for the prediction of a given word are on average shifted by one-third of the expected position compared to the Faster-RCNN predictions. The reason could be related to the fact that, in terms of label prediction, the probabilities given by the object detection module (Faster-RCNN) are not significantly correlated with the notion of importance of the detected objects (regions in the image). In addition, the presence of several features representing the same object can be misleading upon comparison. For instance, in the example of the man playing tennis in Fig. 6, the feature representing *man80%* that is considered the most important by the explanation method for predicting the word *man* in the caption, does not necessarily imply an error in terms of importance rank, simply because *man83%* is considered the most important according to Faster-RCNN.

The ablation experiment based on visual features masking shows similar results using the percentage of missing words as a metric, as shown in Tab. 7. The explanation approaches do not appear to have any advantages over the random baseline, and the LRP-based explanations perform slightly better than the LIME-based ones. One possible reason for these results is the artifact resulting from manipulating individual visual features that often represent partial object rather than entire object (a complete concept). Moreover, the information ablated from the features can be recovered using the rest of the features that correspond to the same object, and this is because the prediction is influenced by the context of the image represented by its visual features. Object-based ablation takes this concern into account, by masking entire objects instead of individual features. The results reported in Tab. 8 show a significant increase in the percentage of missing words and the average probability drop for the different explanation approaches (0.5960 for object oblation against 0.2648 for visual features ablation, for the BU-LIME-5 model), with a notable difference between the proposed methods and the random baseline. The BU-LIME-N model performed best in most cases, while BU-LIME-N-OBJ models that are based on object perturbation rather than individual features perturbation show lower than expected performance. This means that manipulating complete objects instead of isolated visual features is more beneficial for Latent Ablation-based evaluation than for intrinsic use in building explanation models (such as LIME). However, it does not seem to improve the explanation method itself. This leads us to consider the methods based on object manipulation for creating surrogate explanation models as potentially causing gaps in the data leading to inconsistency in the weights of the linear model during the training phase, which could explain the low scores for such models that are often below the random baseline (BU-LIME-5-OBJ and BU-LIME-8-OBJ in Tab. 8). Thus, we believe that object manipulation could instead be used for post-hoc evaluation such as Latent Ablation, but is not recommended for intrinsic use when designing explanatory approaches, since it does not necessarily improve the quality of explanations.

In general, both LRP-based and LIME-based explanations show good quality

compared to the only existing baseline (random) in this perspective of explainability using the latent space. The two techniques seem to be very close to each other in terms of the correctness of their explanations. It turns out that the scope of the explanation technique does not have a significant impact on the explanation quality for the captioning models. This could also depend on how fine-grained an explanation one seeks to obtain. It appears that the global method that back-propagates the relevance of a given prediction until the input is not as instrumental to getting the most important part of the image, while the local method considers taking a shortcut instead of going through the whole model in the opposite direction. It achieves this by considering only the output prediction and linking it to the input via a causal dependency. These results could lead to a greater emphasis on local methods, which are considered easier to achieve in practice than global explainability[1], the latter requiring more complex and time-consuming methods.

Having seen the advantages of both explanation methods, it is now appropriate to discuss their possible shortcomings and limitations. Figure 13 shows a case where the explanations do not seem to be very accurate. The prediction of the word "stop" in the output caption seems to be roughly related to the features representing a "stop sign", except that other features were prioritized by the explanation method, specifically those representing "sky" and "building". In our opinion, these explanation errors could be the result of several facts. First, redundant visual features introduce a kind of bias in the captioning model, making it unclear whether decisions are based on a given feature or its duplicates. In addition, some common concepts such as "sky", "tree", or even "road" are frequent objects and are found in most instances of the dataset. This could lead to explanation biases inherited from captioning biases. However, according to the ablation experiments conducted in the previous section, masking such objects given their explanation importance strongly affects the prediction of the target word ("stop" in this case). Our hypothesis regarding these results is that captioning models are able to build a dependency between concepts during the training phase through the coexistence of certain objects within a scene,

which is then translated into the prediction of a given word from the vocabulary during inference.

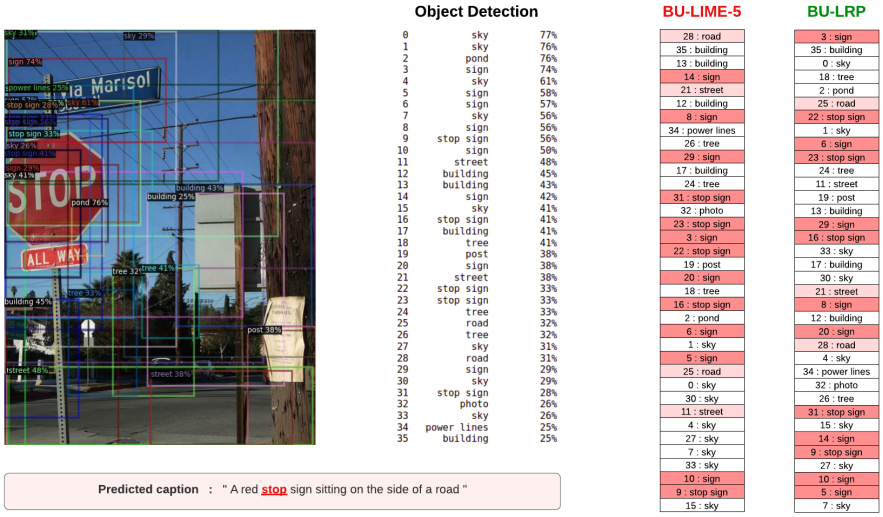

Figure 13: Example of the explanation results generated by BU-LIME-5 and BU-LRP models on a test instance from MSCOCO2017 dataset.

As for future perspectives, it would be interesting to study the possible impact of both the number of extracted features and their quality on the captioning process and the explanation inference. It would also be conceivable to study the factors that may cause bias in captioning architectures, as well as how salient objects are retrieved and distinguished from the background scenes in a given image, things that we believe most influence captioning decisions. Although other explanation approaches exist, in this work we focused on LRP and LIME because of their wide usage in the field, but this does not preclude the possibility of testing our latent space approach on other explanation methods such as Shapley[24] explanations in the future, which might be an interesting lead to consider.

5. Conclusion

In this paper, we propose an end-to-end explainability approach for image captioning (IC) architectures based on Bottom-Up (BU) visual features and the representation space. The first part identifies the most influential component of the IC model for the final caption prediction by injecting Gaussian perturbations into the representation space, while the second one dives deeper into the visual modality that has been shown to be the most decisive component of the captioning architecture. In this context, we adapted two attribution-based explanation methods that are distinct in their scope and functioning, LRP and LIME, to the case of IC and developed a comparative study to assess the quality of the explanations they provide. The two approaches yielded comparative results in terms of the exactitude of their explanations. We also propose the new concept of Latent Ablation for evaluating the quality of explanations. Contrary to classical Ablation, Latent Ablation operates at the latent level of the architecture which allows a better manipulation of the objects in the image and several levels of alteration, thereby enabling a more precise evaluation. Finally, in order to study the impact of concept completeness, i.e. full object handling versus visual features, to perform and evaluate explainability, we designed two versions of LIME and Latent Ablation respectively. The first version is object-based while the second is individual feature-based. The results show that the scope of the explanation method is not as crucial in producing better explanation quality, and that object handling does not tend to improve the robustness of the surrogate explanation model but is a key element in assessing the quality of explanations, as for Latent Ablation. Our code can be found on this GitHub repository.

6. Acknowledgement

This work was supported by the French National Research Agency (ANR) and the University of Orléans, under grant ANR-20-THIA-0017. Our computations were performed using the computing resources of the regional parallel

computing center CaScIModOT provided by the CaScIModOT federation.

References

[1] A. Adadi, M. Berrada, Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI), IEEE Access 6 (c) (2018) 52138–52160. doi:10.1109/ACCESS.2018.2870052.

[2] P. Anderson, B. Fernando, M. Johnson, S. Gould, Spice: Semantic propositional image caption evaluation, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham, 2016, pp. 382–398.

[3] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, W. Samek, On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation, PLOS ONE 10 (7) (2015) 1–46. doi:10.1371/journal.pone.0130140.
URL https://doi.org/10.1371/journal.pone.0130140

[4] S. Banerjee, A. Lavie, METEOR: An automatic metric for MT evaluation with improved correlation with human judgments, in: Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Association for Computational Linguistics, Ann Arbor, Michigan, 2005, pp. 65–72.
URL https://www.aclweb.org/anthology/W05-0909

[5] N. Burkart, M. F. Huber, A survey on the explainability of supervised machine learning, Journal of Artificial Intelligence Research 70 (2021) 245–317.

[6] D. V. Carvalho, E. M. Pereira, J. S. Cardoso, Machine learning interpretability: A survey on methods and metrics, Electronics 8 (8). doi:10.3390/electronics8080832.
URL https://www.mdpi.com/2079-9292/8/8/832

[7] F. K. Došilović, M. Brčić, N. Hlupić, Explainable artificial intelligence: A survey, in: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2018, pp. 0210–0215. doi:10.23919/MIPRO.2018.8400040.

[8] S. Elguendouze, M. C. P. de Souto, A. Hafiane, A. Halftermeyer, Towards explainable deep learning for image captioning through representation space perturbation, in: 2022 International Joint Conference on Neural Networks (IJCNN), 2022, pp. 1–8. doi:10.1109/IJCNN55064.2022.9892275.

[9] R. C. Fong, A. Vedaldi, Interpretable explanations of black boxes by meaningful perturbation, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 3429–3437.

[10] S.-H. Han, H.-J. Choi, Explainable image caption generator using attention and bayesian inference, in: 2018 International Conference on Computational Science and Computational Intelligence (CSCI), 2018, pp. 478–481. doi:10.1109/CSCI46756.2018.00098.

[11] L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, T. Darrell, Generating visual explanations, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham, 2016, pp. 3–19.

[12] S. Herdade, A. Kappeler, K. Boakye, J. Soares, Image captioning: Transforming objects into words, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 32, Curran Associates, Inc., 2019, pp. 11137–11147.
URL https://proceedings.neurips.cc/paper/2019/file/680390c55bbd9ce416d1d69a9ab4760d-Paper.pdf

[13] L. Huang, W. Wang, J. Chen, X.-Y. Wei, Attention on attention for image captioning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 4634–4643.

[14] B. Letham, C. Rudin, T. H. McCormick, D. Madigan, Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model, The Annals of Applied Statistics 9 (3) (2015) 1350 – 1371. doi:10.1214/15-AOAS848.
URL https://doi.org/10.1214/15-AOAS848

[15] G. Li, L. Zhu, P. Liu, Y. Yang, Entangled transformer for image captioning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 8928–8937.

[16] O. Li, H. Liu, C. Chen, C. Rudin, Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions, Proceedings of the AAAI Conference on Artificial Intelligence 32 (1). doi:10.1609/aaai.v32i1.11771.
URL https://ojs.aaai.org/index.php/AAAI/article/view/11771

[17] C.-Y. Lin, ROUGE: A package for automatic evaluation of summaries, in: Text Summarization Branches Out, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 74–81.
URL https://www.aclweb.org/anthology/W04-1013

[18] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, Springer International Publishing, Cham, 2014, pp. 740–755.

[19] J. Liu, H. Jin, G. Xu, M. Lin, T. Wu, M. Nour, F. Alenezi, A. Alhudhaif, K. Polat, Aliasing black box adversarial attack with joint self-attention distribution and confidence probability, Expert Systems with Applications 214 (2023) 119110. doi:https://doi.org/10.1016/j.eswa.2022.119110.
URL https://www.sciencedirect.com/science/article/pii/S0957417422021285

[20] M. Liu, H. Hu, L. Li, Y. Yu, W. Guan, Chinese image caption generation

via visual attention and topic modeling, IEEE Transactions on Cybernetics 52 (2) (2022) 1247–1257. doi:10.1109/TCYB.2020.2997034.

[21] M. Liu, L. Li, H. Hu, W. Guan, J. Tian, Image caption generation with dual attention mechanism, Information Processing & Management 57 (2) (2020) 102178. doi:https://doi.org/10.1016/j.ipm.2019.102178. URL https://www.sciencedirect.com/science/article/pii/S0306457319307885

[22] W. Liu, S. Chen, L. Guo, X. Zhu, J. Liu, Cptr: Full transformer network for image captioning (2021). arXiv:2101.10804.

[23] J. Lu, C. Xiong, D. Parikh, R. Socher, Knowing when to look: Adaptive attention via a visual sentinel for image captioning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 375–383.

[24] S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 30, Curran Associates, Inc., 2017, pp. 4765–4774. URL https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[25] R. Meyes, M. Lu, C. W. de Puiseau, T. Meisen, Ablation studies in artificial neural networks, arXiv preprint arXiv:1901.08644.

[26] Y. Pan, T. Yao, Y. Li, T. Mei, X-linear attention networks for image captioning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 10971–10980.

[27] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, Bleu: a method for automatic evaluation of machine translation, in: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, 2002, pp.

311–318. doi:10.3115/1073083.1073135.
URL https://www.aclweb.org/anthology/P02-1040

[28] G. Ras, N. Xie, M. van Gerven, D. Doran, Explainable deep learning: A field guide for the uninitiated, Journal of Artificial Intelligence Research 73 (2022) 329–397.

[29] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Advances in neural information processing systems 28 (2015) 91–99.

[30] M. T. Ribeiro, S. Singh, C. Guestrin, "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, KDD '16, Association for Computing Machinery, New York, NY, USA, 2016, p. 1135–1144. doi:10.1145/2939672.2939778.
URL https://doi.org/10.1145/2939672.2939778

[31] M. T. Ribeiro, S. Singh, C. Guestrin, Anchors: High-precision model-agnostic explanations, Proceedings of the AAAI Conference on Artificial Intelligence 32 (1). doi:10.1609/aaai.v32i1.11491.
URL https://ojs.aaai.org/index.php/AAAI/article/view/11491

[32] S. Sahay, N. Omare, K. K. Shukla, An approach to identify captioning keywords in an image using lime, in: 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), 2021, pp. 648–651. doi:10.1109/ICCCIS51004.2021.9397159.

[33] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 618–626.

[34] M. Soh, Learning cnn-lstm architectures for image caption generation, Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep.

[35] J. Sun, S. Lapuschkin, W. Samek, A. Binder, Explain and improve: Lrp-inference fine-tuning for image captioning models, Information Fusion 77 (2022) 233–246. doi:https://doi.org/10.1016/j.inffus.2021.07.008.
URL https://www.sciencedirect.com/science/article/pii/S1566253521001494

[36] M. Sundararajan, A. Taly, Q. Yan, Axiomatic attribution for deep networks, in: D. Precup, Y. W. Teh (Eds.), Proceedings of the 34th International Conference on Machine Learning, Vol. 70 of Proceedings of Machine Learning Research, PMLR, International Convention Centre, Sydney, Australia, 2017, pp. 3319–3328.
URL http://proceedings.mlr.press/v70/sundararajan17a.html

[37] E. Tjoa, C. Guan, A survey on explainable artificial intelligence (xai): Toward medical xai, IEEE Transactions on Neural Networks and Learning Systems 32 (11) (2021) 4793–4813. doi:10.1109/TNNLS.2020.3027314.

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 30, Curran Associates, Inc., 2017, pp. 5998–6008.
URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[39] R. Vedantam, C. L. Zitnick, D. Parikh, Cider: Consensus-based image description evaluation, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 4566–4575. doi:10.1109/CVPR.2015.7299087.

[40] T. Wu, X. Wang, S. Qiao, X. Xian, Y. Liu, L. Zhang, Small perturbations are enough: Adversarial attacks on time series prediction, Information Sciences 587 (2022) 794–812. doi:https:

//doi.org/10.1016/j.ins.2021.11.007.
URL https://www.sciencedirect.com/science/article/pii/S0020025521011178

[41] T. Wu, N. Yang, L. Chen, X. Xiao, X. Xian, J. Liu, S. Qiao, C. Cui, Ergcn: Data enhancement-based robust graph convolutional network against adversarial attacks, Information Sciences 617 (2022) 234–253. doi:https://doi.org/10.1016/j.ins.2022.10.115.
URL https://www.sciencedirect.com/science/article/pii/S0020025522012415

[42] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, R. Girshick, Detectron2. 2019, URL https://github. com/facebookresearch/detectron2 2 (3).

[43] X. Xian, T. Wu, S. Qiao, W. Wang, C. Wang, Y. Liu, G. Xu, Deepec: Adversarial attacks against graph structure prediction models, Neurocomputing 437 (2021) 168–185. doi:https://doi.org/10.1016/j.neucom.2020.07.126.
URL https://www.sciencedirect.com/science/article/pii/S0925231220315551

[44] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, Y. Bengio, Show, attend and tell: Neural image caption generation with visual attention, in: F. Bach, D. Blei (Eds.), Proceedings of the 32nd International Conference on Machine Learning, Vol. 37 of Proceedings of Machine Learning Research, PMLR, Lille, France, 2015, pp. 2048–2057.
URL http://proceedings.mlr.press/v37/xuc15.html

[45] X. Xu, X. Chen, C. Liu, A. Rohrbach, T. Darrell, D. Song, Fooling vision and language models despite localization and attention mechanism, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 4951–4961.

[46] Q. Yang, X. Zhu, J.-K. Fwu, Y. Ye, G. You, Y. Zhu, Mfpp: Morphological fragmental perturbation pyramid for black-box model explanations, in:

2020 25th International Conference on Pattern Recognition (ICPR), 2021, pp. 1376–1383. doi:10.1109/ICPR48806.2021.9413046.

[47] Q. You, H. Jin, Z. Wang, C. Fang, J. Luo, Image captioning with semantic attention, in: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), 2016, pp. 4651–4659.

[48] P. Young, A. Lai, M. Hodosh, J. Hockenmaier, From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions, Transactions of the Association for Computational Linguistics 2 (2014) 67–78. arXiv:https://direct.mit.edu/tacl/article-pdf/doi/10.1162/tacl_a_00166/1566848/tacl_a_00166.pdf, doi:10.1162/tacl_a_00166.
URL https://doi.org/10.1162/tacl_a_00166

[49] M. D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, Springer International Publishing, Cham, 2014, pp. 818–833.

[50] R. Zellers, M. Yatskar, S. Thomson, Y. Choi, Neural motifs: Scene graph parsing with global context, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 5831–5840.

[51] S. Zhang, Z. Wang, X. Xu, X. Guan, Y. Yang, Fooled by imagination: Adversarial attack to image captioning via perturbation in complex domain, in: 2020 IEEE International Conference on Multimedia and Expo (ICME), 2020, pp. 1–6. doi:10.1109/ICME46284.2020.9102842.

[52] Y. Zhong, L. Wang, J. Chen, D. Yu, Y. Li, Comprehensive Image Captioning via Scene Graph Decomposition, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 12359 LNCS, Springer Science and Business Media Deutschland GmbH, 2020, pp. 211–229. arXiv:2007.11731,

doi:10.1007/978-3-030-58568-6_13.

URL https://link.springer.com/chapter/10.1007/978-3-030-58568-6_13