A General Approach to Compute the Relevance of Middle-Level Input Features

Andrea Apicella(✉), Salvatore Giugliano, Francesco Isgrò, and Roberto Prevete

Dipartimento di Ingegneria Elettrica e delle Teconologie dell'Informazione,
Università degli Studi di Napoli Federico II, Naples, Italy
and.api.univ@gmail.com

Abstract. This work proposes a novel general framework, in the context of eXplainable Artificial Intelligence (XAI), to construct explanations for the behaviour of Machine Learning (ML) models in terms of middle-level features which represent perceptually salient input parts. One can isolate two different ways to provide explanations in the context of XAI: low and middle-level explanations. Middle-level explanations have been introduced for alleviating some deficiencies of low-level explanations such as, in the context of image classification, the fact that human users are left with a significant interpretive burden: starting from low-level explanations, one has to identify properties of the overall input that are perceptually salient for the human visual system. However, a general approach to correctly evaluate the elements of middle-level explanations with respect ML model responses has never been proposed in the literature.

We experimentally evaluate the proposed approach to explain the decisions made by an Imagenet pre-trained VGG16 model on STL-10 images and by a customised model trained on the JAFFE dataset, using two different computational definitions of middle-level features and compare it with two different XAI middle-level methods. The results show that our approach can be used successfully in different computational definitions of middle-level explanations.

Keywords: XAI · Machine Learning · Middle-level features

1 Introduction

In the last years, Machine Learning (ML) approaches have been widely used to address several challenges in Artificial Intelligence (AI), such as image [31] and text classification [11] problems, multi-target regression [25] and robot navigation [28]. However, a large part of these approaches suffers from a pervasive lack of transparency also connected to the problem of explaining their behaviour in terms that are easy to understand for human beings [18]. Indeed, it seems that the better ML systems become in terms of their performance, the harder it is to understand the underlying mechanisms and explain their behaviours [1]. For this reason, ML systems are often considered as black-box systems [1] insofar as

A. Del Bimbo et al. (Eds.): ICPR 2020 Workshops, LNCS 12663, pp. 189–203, 2021.
https://doi.org/10.1007/978-3-030-68796-0_14

their decisions are hard to interpret in terms of meaningful input features. Thus, generating explanations for ML system behaviours that are *understandable to human beings* is a central scientific and technological issue addressed by the rapidly growing AI research area of eXplainable Artificial Intelligence (XAI).

The literature counts various strategies to make ML systems - especially those endowed with Deep Neural Network (DNN) architectures [21] - interpretable and explainable [12,22]. XAI approaches to the explanation problem can be classified in several ways according to which properties are taken into account [1,15,23,34]. A key distinction is between *low-level* and *middle-level* input feature approaches. Low-level feature approaches to XAI attempt to explain the output of an ML system in terms of low-level features of the input such as pixels in case of image classification problems. One of the most successful methods for this type of approaches is the Layer-wise Relevance Propagation (LRP) [6], which associates a relevance value to each input element (to each pixel in the case of images) as an explanation of the ML model response. Thus, human users are left with a significant interpretive burden: starting from the relevance values of each input element (pixel), one has to identify properties of the overall input that are perceptually salient for the human visual system. A method which attempt to alleviate this drawback of low-level approaches to explanation have been proposed in [3,4], where explanations are provided in terms of middle-level properties (*atoms*) of the input which represent perceptually salient input parts [7].

A popular method which is also based on middle-level properties of the input is LIME [26], which returns a set of image parts (*superpixels*), that could have driven the ML model to the given answer. This set of superpixels can be then considered as an explanation to the ML model response. To the best of our knowledge, these types of approaches can be classified as *model-agnostic*.

Model-agnostic approaches correspond to XAI methods which are independent of the ML model to be explained [1], i.e., model-agnostic solutions are built relying only the relation between ML model inputs and outputs, without any consideration about the ML model internal state. Although this property ensures the applicability of these approaches to any ML model, on the other hand, how we will discuss more in details in Sect. 3, the explanations of the model-agnostic methods could not be fully related to the actual causal relationships between model's inputs and outputs which have contributed to the given model response. For instance, LIME returns an explanation inspecting the behaviour of the model in the neighbourhood of the input, but nothing ensures that, for that particular input instance, the answer of the classifier has a totally different explanation (for example, a particular on the background of the specific input image which the model has already seen during the training stage, making the model biased).

In this paper, we propose a new method, that we called Middle-Level Feature Relevance (MLFR), based on a variation of LRP that, instead of returning a relevance value for each input pixel, returns relevance values for a given set of middle-level features. This method can be applied whenever a) the input of a ML

system can be encoded and decoded on the basis of middle-level features, and b) LRP can be applied on both the ML model and the decoder (see Sect. 3 for further details). In this sense we consider MLFR as a *a general framework* insofar as it can be applied on several different computational definitions of middle-level features as we will discuss in Sect. 3. Notice that MLFR is not a model-agnostic approach, however it can be applied to a large class of ML models as well as LRP [6], for example feedforward neural networks architectures such as shallow network and deep networks.

This paper is structured as follows. Section 2 briefly reviews the related literature; Section 3 describes the proposed architecture; experiments and results are discussed in Sect. 4; the concluding Sect. 5 summarises the main results of the proposed explanation framework and outlines some future developments.

2 Related Works

Many XAI methods have been proposed since explainability is now a sought for requirement for AI solution. The literature proposes several reviews trying to categorise/distinguish the existing methods [1,15,21,34] looking at different properties of the XAI methods. According to these categorisations, our method can be classified as a *white-box* and *local* XAI approach. White-box approaches require access to the internal structure of the ML model [1]. By contrast, black-box, or *model-agnostic*, approaches provide explanation methods which are independent of the ML model [1], i.e., they need access only to the input-output relations of the ML model. Local approaches provide explanations for each given input, while the goal of global approaches is to produce an explanation for the whole behaviour of the ML system [21].

Many model-agnostic approaches are based on *proxy-models* [8,10,24] or some type of maximisation of the ML model response with respect to the input, such as the Activation-Maximisation (AM) method [12]. Proxy models are models behaving similarly to the original model, but in a way that it is easier to explain [13]. Approaches based on AM method enables one to determine the input that makes the output of the ML model as close as possible to the model's initial response, for example, in case of classification problems, given C_k as the response of the ML model, one maximises the $P(C_k|\mathbf{x})$ with respect to $\mathbf{x}$ satisfying some constraints on $\mathbf{x}$. Notice that the explanations of the model-agnostic methods suffer from the lack of information about the actual input-output causal relationships which have contributed to the given ML model answers; thus these explanations may not be related to the specific ML model response to be explained [19].

Another critical distinction is based on the granularity level of the explanations. In fact, several XAI solutions provide explanations in terms of low-level input features. For instance, in image classification problems the output of an ML system is explained considering low-level features of the input image in terms of salience maps where to each pixel is associated a relevance value which quantifies the degree of importance of that pixel to cause the ML model response.

Among the approaches of this type, Layer-wise Relevance Propagation (LRP) [6], is the most popular in the literature. LRP is a white-box approach, although it applies to many ML models such as deep networks. Notice that it is a general framework rather than a specific method insofar as it is defined as a set of constraints that an XAI algorithm should satisfy. Thus, different XAI algorithms with different explanations may be appropriate under these constraints [6]. For example, Deep-Taylor Decomposition [21] can be interpreted as a way of obtaining LRP.

In this type of approaches, human users are left with a significant interpretive burden: starting from the relevance values of each input element (pixel), one has to identify properties of the overall input that are perceptually salient for the human visual system. Thus, to alleviate this cognitive burden, an alternative model-agnostic method, called Explanation-Maximization (EM), was proposed in [2–4]. EM, which also applies in different areas, was instantiated in the context of image classification systems. EM obtains sets of perceptually salient middle-level properties of input images by applying sparse dictionary learning techniques and a variant of AM. These middle-level properties are used as building blocks for explanations of image classifications. However, this approach suffers from the typical shortcomings of the model-agnostic ones as regards the reliability of the explanations given, as previously discussed. Among other methods using middle-level input features to build explanations about ML model responses, LIME [26] can be considered the most popular in the literature. It is model-agnostic and based on a proxy-model: it explains the output of an ML system by observing its behaviour on perturbations of its input. The input is partitioned in a collection of *components* (super-pixel in the case of images); perturbed inputs are composed of specific superpositions of these components. Perturbed inputs and outputs are used to construct a local linear model which is used as a simplified proxy for the original ML system in the neighbourhood of the input. Thus, from the proxy, it is possible to infer an explanation of the original ML model response. However, the faithfulness of the proxy with respect to the original model remains an open issue [19]. Other methods, based on LIME, as in Ribeiro et al. [27] and Guidotti et al. [14], return explanations in terms of decision rules that are used as local conditions for decisions.

The method we propose in this paper differs from the works mentioned above in the following aspects, as it can be seen as a general framework to obtain middle-level explanations analysing the actual input-output relationship defined by the ML model. Thus, different definitions of middle-level input features with different resulting solutions may be possible under this general framework. The only constraints are that the input can be encoded and decoded based on the defined middle-level input features and that LRP can be applied on both the ML model to be explained and the input decoder.

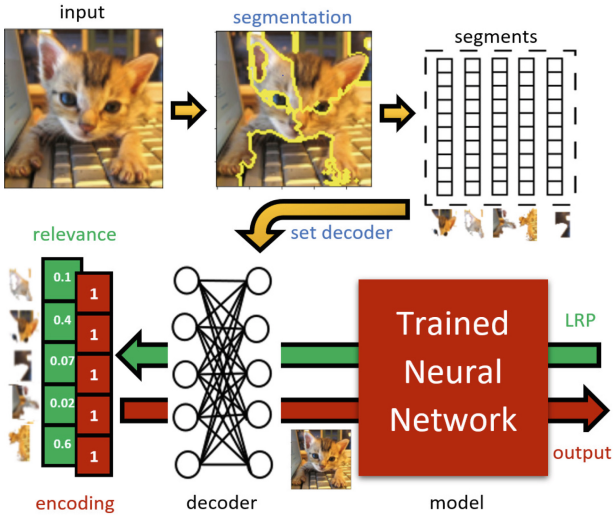
3 Middle-Level Relevance

Given an ML model M which receives an input $\mathbf{x} \in R^d$ and outputs $\mathbf{y} \in R^c$, let us suppose that $\mathbf{x}$ can be decomposed in a set of m middle-level features

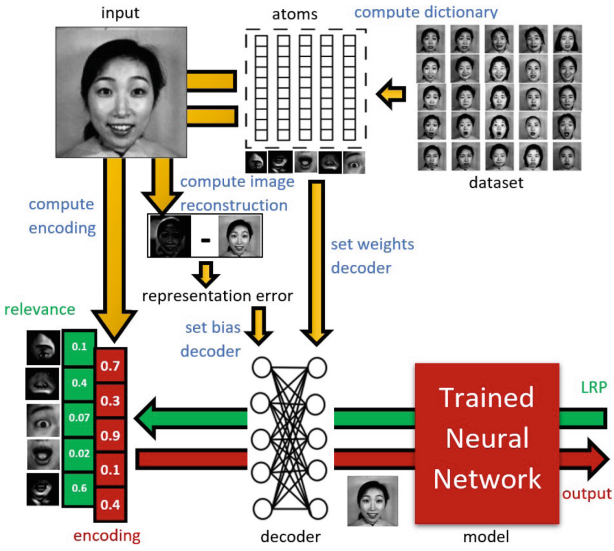
$\mathbf{v}^i$ each one encoded by a value u_i. More formally, we suppose that a decoder $D : (\mathbf{V}, \mathbf{u}) \rightarrow \mathbf{x} \in R^d$ exists. Where $\mathbf{V} = \{\mathbf{v}^i\}_{i=1}^m$ is the set of $\mathbf{v}$'s middle-level features and $\mathbf{u} \in R^m$ encodes $\mathbf{x}$ in terms of the middle-level features. For example, in an image classification problem, a possible set of middle-level features can be the result of a segmentation algorithm on the input image $\mathbf{x}$ which produces a partition of $\mathbf{x}$ in m regions or partitions $\{P_i\}_{i=1}^m$. Each image's partition P_i can be represented by a vector $\mathbf{v}^i \in R^d$ such that their summation is equal to $\mathbf{x}$, in this case the decoder is a linear combination of the $\mathbf{v}^i$ with all the coefficients equal to 1, which represent the encoding of the image $\mathbf{x}$ on the basis of the m partitions (see Sect. 3.1).

Then, if we can use LRP on both M and D, we can apply it on the model M and use the obtained relevance values to apply LRP on D thus getting a relevance value for each middle-level feature. In other words, we can stack D on the top of M, thus obtaining a new model DM which receives as input $\mathbf{u}$ and outputs $\mathbf{y}$, and uses LRP propagation on DM from $\mathbf{y}$ to $\mathbf{u}$. Let us take as an example of M a neural network composed of L layers. The LRP procedure computes a set of relevance values for any given layer l composed of k_l neurons as the combination of the scores assigned to each neuron of l, representing the importance of each node for the network's output. The scores are computed by propagating the relevance values from the output layer to the input layer in a back-propagation fashion. Similarly, let us consider a shallow neural network composed of m input values u_i, d output neurons with biases equal to 0, the identity as activation functions, and one hidden layer of weights $W = V$. This network can be seen as a decoder D where the weights associated with the connections going from each input value u_i to all output neurons represent the middle-level feature vector $\mathbf{v}^i$. If we stack the shallow network/decoder D on the top of the L-layer model M, we obtain a new neural network model DM composed of $L + 1$ layers. Then, we can apply the LRP procedure on the whole DM model and obtain relevance values which designate what input's middle-level features have most contributed on the outcome y_i (see Fig. 1). In other words, we search for a relevance vector $\mathbf{r} \in \mathbb{R}^m$ which helps the user to know each middle-level feature of $\mathbf{x}$ how much has contributed to the ML model answer y_i. Note that, this approach can be generalised to any decoder to which LRP applies. For instance, we can consider any dictionary learning approach, as for example [16,17,32] (see Sect. 3.2 for more details), where each input $\mathbf{x}$ can be decomposed as $\mathbf{x} = \mathbf{V}\mathbf{u} + \epsilon$, V is a dictionary of middle-level features and ϵ is the reconstruction error vector. Also, in this case, we can notice that the decoder can be represented as a shallow neural network having the dictionary elements as weights and the biases in terms of the reconstruction error vector (see Sect. 3.2).

In the remainder of this section, we will describe two alternative ways (segmentation and dictionary learning) to obtain a decoder LRP method can be applied to, in more details. We experimentally tested our framework using both methods.



(a) The segmentation-based approach.



(b) The dictionary-based approach.

Fig. 1. A description of the proposed method (MLFR) using two different types of middle-level features. (a) After segmenting the input, the segments are used as weights for the decoder, so feeding the decoder with the **1**s is equivalent to give the input image to the trained neural network. After, the LRP algorithm is used to obtain the segment relevances (see Sect. 3.1 for further details). (b) Having a dictionary, and an input encoding which best approximates the input image, we can use the dictionary and the representation error respectively as weights and bias of the decoder. So, feeding the decoder with the input encoding is equivalent to give the network the input image. After, the LRP algorithm is used to obtain the atom relevances (see Sect. 3.2 for further details).

3.1 Decoder by Super-Pixel Segmentation

Given an image $\mathbf{x} \in R^d$, we can obtain a partition of $\mathbf{x}$ composed of m elements P_h through any segmentation algorithm. We can associate to each element P_h a vector $\mathbf{pv}^h \in R^d$ such that $pv_i^h = 1$ if $x_i \in P_h$, otherwise $pv_i^h = 0$. Thus, each element P_h can be represented by the element-wise product between $\mathbf{x}$ and $\mathbf{pv}^h$, i.e., $\mathbf{v}^h = \mathbf{pv}^h \odot \mathbf{x}$, since this operation products selects all the pixels belonging to the element P_h.

Consequently, we can decompose $\mathbf{x}$ as $\mathbf{x} = \sum_{h=1}^{m} u_h \mathbf{v}^h$, with $u_h = 1$. Then, the decoder D is a linear combination of the $\mathbf{v}^h$ with all the coefficients equal to 1, which represent the encoding of the image $\mathbf{x}$ on the basis of the m partition's elements.

Following [26], in this paper we use the Quickshift segmentation algorithm [33] where the elements of the partition are called super-pixels.

We assume that a possible explanation to the output of a given classifier can be obtained in terms of relevant super-pixels, where the relevance can be computed using an LRP-based procedure.

3.2 Decoder by Sparse Dictionary Learning Methods

A sparse dictionary learning problem (see, for example, [32]) is a minimisation problem that one can formally describe as follows.

$$\arg\min_{U,V} ||X - VU||_F^2 + \gamma_1 \sum_{i=1}^{k} \Omega_V(\mathbf{v}_i) \quad \text{s.t. } \forall i, \Omega_U(\mathbf{u}_i) < \gamma_2 \tag{1}$$

where $X \in R^{d \times n}$ is composed of n experimental observations which are expressed as column vector $\mathbf{x}^i \in R^d$, V is the dictionary, and the k columns $\mathbf{v}^i$ of V are the dictionary elements or atoms, subject to some sparsity constraint possibly. Each column of X is approximated by a linear combination of the k columns of V, subjects to some sparsity constraint potentially. Thus, $U \in R^{k \times n}$ is the matrix of the linear combination coefficients, i.e., the i-th column of U, $\mathbf{u}^i$, corresponds to the k coefficients of the linear combination of the k columns of V to approximate $\mathbf{x}^i$, the i-th column of X. Ω_V and Ω_U are some norms or quasi-norms that constrain or regularise the solutions of the minimisation problem, and $\gamma_1 \geq 0$ and $\gamma_2 \geq 0$ are parameters that control to what extent the dictionary and the coefficients are regularised.

Elements of a dictionary can be used to compute explanations of a ML model response in terms of middle-level input features [2–5].

For the experiments presented in this paper, we obtain the dictionaries from a specific sparse dictionary learning method based on SSPCA [16]. However, any dictionary learning/sparse coding method able to produce dictionaries that can be considered human-understandable can be used [2].

Given a dictionary V and an experimental observation $\mathbf{x}$ one can solve the minimisation problem as expressed in Eq. 1 with respect to the coefficients only, finding a single column vector $\mathbf{u}$. Consequently, $\tilde{\mathbf{x}} = V\mathbf{u}$ is an approximation of $\mathbf{x}$ with an error for each component equal to $\epsilon_h = x_h - \tilde{x}_h$. Then, the decoder D can be represented as a shallow neural network composed of just one weight layer W, k input values and d output neurons. Each output neuron j has the identity as activation function and the bias equal to ϵ_j. The weights associated to the connection going from the j-th input value to all the output neurons correspond to j-th V's column. Consequently, given the decomposition of $\mathbf{x}$ as $V\mathbf{u}$ the decoder D receives $\mathbf{u}$ as input and outputs $\mathbf{x}$.

4 Experimental Assessment

In this section, we describe the experiments performed and show the results obtained. We show a set of explanations produced by our approach using two different experimental setups.

The former uses as middle-level features the super-pixel segmentation schema described in Sect. 3.1; the latter adopts the sparse dictionary approach described in Sect. 3.2. For the segmentation-based experiments, we used as classifier a VGG-16 [30] neural network trained on Imagenet, and as input images a subset of the STL-10 dataset [9]. For the dictionary-based experiments, we use the JAFFE dataset [20] and a custom neural network trained from scratch with a final accuracy of the 94% on a test set. We chose to use a custom model because, to the best of our knowledge, there are no reference models for this particular dataset in the current literature. Notice that for this type of middle-level features we used a more simple dataset since dictionary learning methods on large datasets can be very expensive in terms of computational costs.

We compare the results obtained by the proposed method (MLFR) with two related methods proposed in the literature, LIME [26] and EM [2,4], and with a standard low-level feature method as LRP [6]. Notice that as we discussed in Sect. 2 the explanations returned by LIME and EM are based on features which can be considered of middle-level, but, differently from the MLFR approach, they are built in a black-box approach relying on a proxy model instead of the actual model in case of LIME, and in terms of dictionary elements by a variant of the activation-maximisation method, in case of EM. For the segmentation-based approach, we compared MLFR with LIME and LRP. For the dictionary-based approach, we compared our results with the ones produced by EM and LRP. The segments and the dictionaries are obtained respectively using Quickshift [33] (that is the same algorithm used by LIME to make the superpixel segmentation) and SSPCA [16].

A visual comparison is not enough, and to give a quantitative evaluation of our results, we use the same strategy introduced in [29] and described in Sect. 4.2.

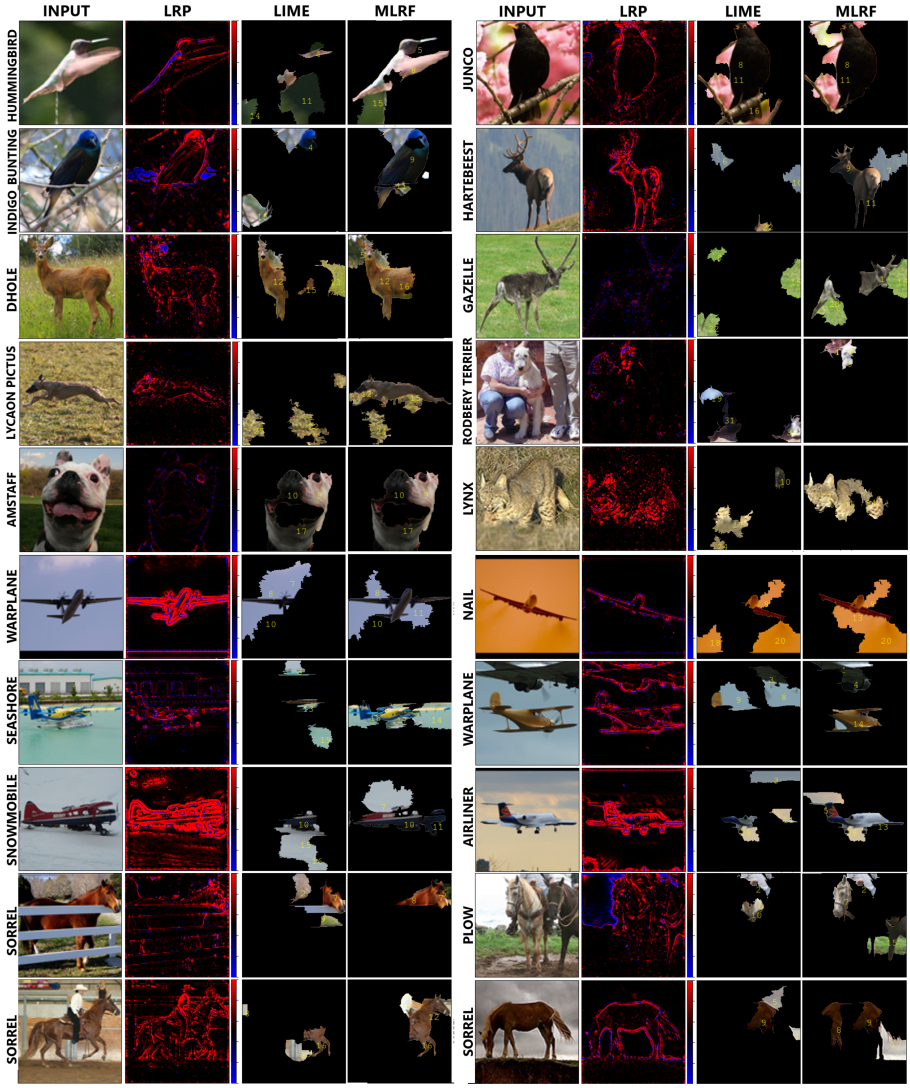


Fig. 2. Results obtained from MLFR using image segmentation (Sect. 3.1) on the STL10 dataset. For each input (1st and 5th column), we present the explanations produced by LRP method (2nd, 6th column) in terms of heatmaps (blue pixels indicate negative relevance, while red pixels indicate positive ones), LIME method (3th and 7th column) and MLFR (4rd and 8th column) as superimposition of the three superpixels with the highest relevance scores. The class returned by the classifier is reported for each input. (Color figure online)

Fig. 3. Results obtained from MLFR with sparse dictionaries (Sect. 3.2) on the JAFFE dataset. For each input (1st, 4th and 7th column), we present the explanations obtained by EM method (2nd, 5th and 8th column) and MLFR (3rd, 6th and 9th column) as superimposition of the three atoms with the highest relevance scores. On the left of each input, we report the class returned by the classifier.

4.1 Qualitative Results

Some results of the two proposed strategies are shown in Fig. 2 for the superpixels-based approach and in Fig. 3 for the dictionary-based approach. To make a comparison, we also report the explanations given by LIME and LRP methods for the superpixels-based approach and EM for the dictionary-based

approach. For each input, we show the superposition of the three most relevant segments/atoms for LIME, EM and MLFR, and the heatmap produced by LRP.

With respect to LIME and EM, we can show that in several cases, the explanations produced by MLFR can be considered closer to what a human being expects from a classification system. For example, we expect that the output "hummingbird" and "indigo bunting" (Fig. 2, first and second row, first column) is due mainly by the presence of the main components of a bird in the image, neglecting non-relevant part as background sprigs. Similar considerations can be done for the "hartebeest" and the "gazelle" (Fig. 2, second and third row, fifth column) and several other inputs shown in the figure. In particular, the "Bedlington Terrier" input (Fig. 2, fourth row, fifth column) provides an interesting case due to the presence of several hypothetical relevant candidates (the several human being parts) which can lead the classifier toward different classification. The proposed method, in agreement with LRP, highlights that the dog face is one of the main discriminative parts behind the classifier's choice. The different results returned by LIME can be due to several factors, such as a sub-optimal training procedure of the proxy model or the inadequacy of the proxy model in representing the real one.

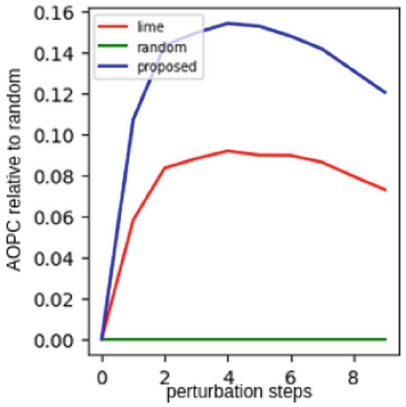
Similar consideration can be done for the results shown in Fig. 3 inherent the dictionary-based approach. We show inputs for several classes of the Jaffe dataset (SAD, SURPRISE, HAPPY, FEAR, ANGRY) and the results obtained respectively by the EM method and the proposed MLFR. It is possible to see how the proposed method highlights atoms which better characterise the faces, as we would expect by an emotion classifier. MLFR highlights details concerning facial expressions as the open mouth and the eyes for the inputs classified as "SURPRISE" or the smiling expression for the input classified for "HAPPY" (for example, on the Fig. 3 see the results of the inputs on the first, fourth and seventh columns of the 5th and 8th row). The results produced by EM method, instead, seems more confused and less clear and intuitive. As far as the EM method is not based on proxy models, it is again a black-box approach based only on the input/output relations of the classifier, so without any knowledge to the real inner state of the model. Furthermore, EM needs several hyperparameters to be set, which can affect the reliability of the results produced.

Notice that for reasons of space we do not report the results obtained by LRP, since the qualitive comparison is similar to the one for the STL10 dataset.

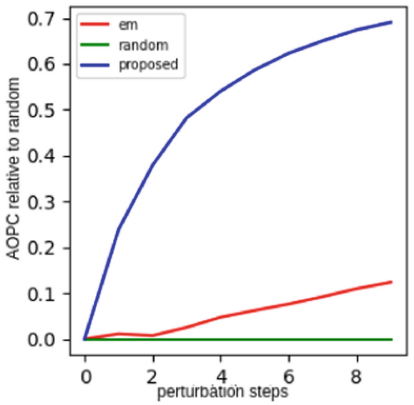
4.2 Quantitative Evaluation

In the previous section, we show the explanations obtained in terms of the most relevant middle-level features selected by MLFR compared against the ones selected by some related works proposed in the literature. However, all the consideration we made are based only on subjective evaluations, and an objective and quantitative evaluation of the explanation methods is still an open research problem.

A possible quantitative evaluation framework was proposed in [29] with *region flipping*, a generalisation of the *pixel-flipping* measure proposed in [6]. In a nutshell, given an image classification to explain, regions of a given size are



(a) AOPC curve of the proposed method (segmentation-based, section 3.1) compared with the LIME method.



(b) AOPC curve of the proposed method (DL-based, section 3.2) compared with the EM method.

Fig. 4. Comparison of the AOPC curves of the methods used in the experiments. As made in [29], all the curves have been plotted relatively to a random AOPC curve, which was obtained following a random order instead of a relevance order during the image perturbation steps.

substituted iteratively, following the descending relevance order assigned to the central pixel (MoRF, Most Relevant First) by the explanation method. At each step, the difference between the original class score returned by the model and the score returned on the perturbed input is computed, generating a curve (MoRF curve). We expect that the better the explanation method is, the stronger the difference between the scores is. Repeating this process for several images and averaging between them, it is possible to obtain the *Area Over the MoRF Perturbation Curve* (AOPC):

$$AOPC = \frac{1}{L+1} < \sum_{k=0}^{L} f(x^{(0)}) - f(x^{(k)}) >_{p(x)}$$

where $< \cdot >_{p(x)}$ is the average over the dataset images, L is the number of regions and $x^{(k)}$ is the input at $k-$th perturbation step. If the regions are well-ranked (so, relevant regions have a higher relevance), we expect that the resulting AOPC values are large, so we can infer that the largest the AOPC value is, the better the explanation method is. The original region-flipping method was originally defined for pixel-based heatmaps using regions of fixed size (9×9 in [29]). However, it is easily adapted to our proposed method and LIME, considering that each middle-level feature is a single region. As a perturbation scheme, we adopt the same used in [29], changing each pixel in the region with a value sampled from the Uniform distribution. In Fig. 4a we plot the AOPC curve for LIME and our proposed method on the VGG16 model, showing that MLFR outperforms LIME in terms of AOPC curve, suggesting that the former, on average, gives a

more reliably relevance score respect to the latter. We hypothesise that LIME, exploiting a proxy classifier which *emulates* the real one, may not capture the real "reasons" behind the choices made by a classifier, so assigning scores to the features in a manner which not reflect the real inner state of the classifier. Similar results are shown in Fig. 4b, where the results of the proposed approach are compared with the EM method. Again, in this case, the proposed method shows better results in terms of AOPC values, giving better reliability to the explanations produced.

5 Conclusions

In this work, we propose MLFR, a novel XAI method based on middle-level features. The proposed method generalises the well-known LRP method, initially proposed for low-level features (such as pixels for image domain), to middle-level features, returning data representations which can be interpreted by a human. We describe how the proposed method can be easily adapted to several classes of middle-level features. For instance, we show how two different middle-level input representations can be suitable for the proposed method, the former based on image segments directly obtained from the input to explain, the latter on a more general set of elements which can be constructed through some dictionary learning approach. However, nothing prevents to use other representations.

To evaluate the proposed method, we adapt the quantitative measure described in [29], proposed initially for pixelwise-based methods, to middle-level feature methods, and we make a comparison with others middle-level features approaches present in literature. The results of the experiments that we carried out are encouraging, both under the qualitative point of view, giving explanations that can be easily interpretable by the human being, and the quantitative point of view, giving performances in terms of AOPC curve which are comparable to other methods present in the current literature.

Acknowledgments. The research presented in this paper was partially supported by the national project Perception, Performativity and Cognitive Sciences (PRIN Bando 2015, cod. 2015TM24JS 009).

References

1. Adadi, A., Berrada, M.: Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). IEEE Access **6**, 52138–52160 (2018)
2. Apicella, A., Isgro, F., Prevete, R., Sorrentino, A., Tamburrini, G.: Explaining classification systems using sparse dictionaries. In: Proceedings of the ESANN, Special Session on Societal Issues in Machine Learning: When Learning from Data is Not Enough. Bruges, Belgium (2019)
3. Apicella, A., Isgrò, F., Prevete, R., Tamburrini, G.: Contrastive explanations to classification systems using sparse dictionaries. In: Ricci, E., Rota Bulò, S., Snoek, C., Lanz, O., Messelodi, S., Sebe, N. (eds.) ICIAP 2019, Part I. LNCS, vol. 11751, pp. 207–218. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30642-7_19

4. Apicella, A., Isgrò, F., Prevete, R., Tamburrini, G.: Middle-level features for the explanation of classification systems by sparse dictionary methods. Int. J. Neural Syst. **30**(08), 2050040 (2020)
5. Apicella, A., Isgro, F., Prevete, R., Tamburrini, G., Vietri, A.: Sparse dictionaries for the explanation of classification systems. In: PIE, p. 009. Rome, Italy (2019)
6. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS One **10**(7), e0130140 (2015)
7. Barghout, L.: Spatial-taxon information granules as used in iterative fuzzy-decision-making for image segmentation. In: Pedrycz, W., Chen, S.-M. (eds.) Granular Computing and Decision-Making. SBD, vol. 10, pp. 285–318. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16829-6_12
8. Caccavale, R., Finzi, A.: Learning attentional regulations for structured tasks execution in robotic cognitive control. Auton. Robot. **43**(8), 2229–2243 (2019). https://doi.org/10.1007/s10514-019-09876-x
9. Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, pp. 215–223 (2011)
10. Craven, M., Shavlik, J.W.: Extracting tree-structured representations of trained networks. In: Advances in Neural Information Processing Systems, pp. 24–30. Denver, CO, USA (1996)
11. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding (2018)
12. Erhan, D., Bengio, Y., Courville, A., Vincent, P.: Visualizing higher-layer features of a deep network. Univ. Montreal **1341**(3), 1 (2009)
13. Gilpin, L.H., Bau, D., Yuan, B.Z., Bajwa, A., Specter, M., Kagal, L.: Explaining explanations: an overview of interpretability of machine learning. In: 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), pp. 80–89. IEEE, Turin, Italy (2018)
14. Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., Giannotti, F.: Local rule-based explanations of black box decision systems. CoRR abs/1805.10820 (2018)
15. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., Pedreschi, D.: A survey of methods for explaining black box models. ACM Comput. Surv. (CSUR) **51**(5), 93 (2018)
16. Jenatton, R., Obozinski, G., Bach, F.: Structured sparse principal component analysis. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp. 366–373 (2010)
17. Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: Advances in Neural Information Processing Systems, pp. 556–562 (2001)
18. Letham, B., Rudin, C., McCormick, T.H., Madigan, D., et al.: Interpretable classifiers using rules and Bayesian analysis: building a better stroke prediction model. Ann. Appl. Stat. **9**(3), 1350–1371 (2015)
19. Li, X.H., et al.: A survey of data-driven and knowledge-aware explainable AI. IEEE Trans. Knowl. Data Eng. (2020). https://doi.org/10.1109/TKDE.2020.2983930
20. Lyons, M., Akamatsu, S., Kamachi, M., Gyoba, J.: Coding facial expressions with Gabor wavelets. In: Proceedings, Third IEEE International Conference on Automatic Face and Gesture Recognition, pp. 200–205. IEEE Computer Society (1998)
21. Montavon, G., Samek, W., Müller, K.: Methods for interpreting and understanding deep neural networks. Digit. Signal Process. **73**, 1–15 (2018)

22. Nguyen, A., Dosovitskiy, A., Yosinski, J., Brox, T., Clune, J.: Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29, pp. 3387–3395. Curran Associates, Inc. (2016)
23. Nguyen, A., Yosinski, J., Clune, J.: Understanding neural networks via feature visualization: a survey. In: Samek, W., Montavon, G., Vedaldi, A., Hansen, L.K., Müller, K.-R. (eds.) Explainable AI: Interpreting, Explaining and Visualizing Deep Learning. LNCS (LNAI), vol. 11700, pp. 55–76. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-28954-6_4
24. Oh, S.J., Schiele, B., Fritz, M.: Towards reverse-engineering black-box neural networks. In: Samek, W., Montavon, G., Vedaldi, A., Hansen, L.K., Müller, K.-R. (eds.) Explainable AI: Interpreting, Explaining and Visualizing Deep Learning. LNCS (LNAI), vol. 11700, pp. 121–144. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-28954-6_7
25. Reyes, O., Ventura, S.: Performing multi-target regression via a parameter sharing-based deep network. Int. J. Neural Syst. **29**, 1950014 (2019)
26. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should i trust you?": Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144. KDD '16, ACM (2016)
27. Ribeiro, M.T., Singh, S., Guestrin, C.: Anchors: High-precision model-agnostic explanations. In: Thirty-Second AAAI Conference on Artificial Intelligence. New Orleans, Louisiana, USA (2018)
28. Richter, C., Vega-Brown, W., Roy, N.: Bayesian learning for safe high-speed navigation in unknown environments. In: Bicchi, A., Burgard, W. (eds.) Robotics Research. SPAR, vol. 3, pp. 325–341. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-60916-4_19
29. Samek, W., Binder, A., Montavon, G., Lapuschkin, S., Müller, K.R.: Evaluating the visualization of what a deep neural network has learned. IEEE Trans. Neural Netw. Learn. Syst. **28**(11), 2660–2673 (2016)
30. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (2015)
31. Springenberg, J., Dosovitskiy, A., Brox, T., Riedmiller, M.: Striving for simplicity: the all convolutional net. In: Proceedings of the International Conference on Learning Representation (Workshop Track). San Diego, CA (2015)
32. Tessitore, G., Prevete, R.: Designing structured sparse dictionaries for sparse representation modeling. In: Burduk, R., Kurzynski, M., Wozniak, M., Zolnierek, A. (eds.) Computer Recognition Systems 4. Advances in Intelligent and Soft Computing, vol. 95, pp. 157–166. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20320-6_17
33. Vedaldi, A., Soatto, S.: Quick shift and kernel methods for mode seeking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 705–718. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88693-8_52
34. Zhang, Q., Zhu, S.: Visual interpretability for deep learning: a survey. Front. Inf. Technol. Electron. Eng. **19**(1), 27–39 (2018). https://doi.org/10.1631/FITEE.1700808