

TECHNICAL UNIVERSITY OF MUNICH

GUIDED RESEARCH

Summer Semester 2022

Explainable AI for clinical decision support in dermatology

Supervisor: PD Dr. rer. nat. Tobias Lasser

Advisor: Theodor Cheslerean-Boghiu, MSc

Student: Aydin Javadov

Munich, 20.10.2022

Table of Contents

Abstract	3
1. Introduction.....	3
1.1. Problem Statement	4
1.2. Fundamentals of XAI – Literature	4
1.2.1. Interpretability vs. Explainability	4
1.2.2. Core Concepts	4
1.2.3. Evaluation methods of XAI	7
1.3. Refined Problem View and Our Approach	7
2. Methods.....	9
2.1. Data and Model Set-up	9
2.2. Learning	9
2.2.1. Seven-Point Checklist dataset learning	9
2.2.2. ISIC-2019 dataset learning.....	10
2.3. Explainable AI	12
2.3.1. Explaining deep learning models	12
2.3.2. Explaining any black-box model.....	17
3. Conclusion and Future Work	22
REFERENCES	23

Abstract

Continuous advancements and remarkable results in the Artificial Intelligence (AI) sphere inspire a number of domains to incorporate AI techniques into their routines as well. The concept of Explainable AI (XAI) becomes especially vital when the domains involve sensible topics, such as cases where important decisions for humans must be taken. The medical context, in our case the dermatological domain, is a suitable example of a situation when the performance of AI should be justifiable with a fair enough XAI performance, for a number of reasons, such as ethical and legal perspectives. On the other hand, skin lesion diagnosis, which may potentially save lives, especially if there is melanoma involved, is of crucial importance in dermatology. This work first presents common XAI concepts and methodologies with an extensive literature review, followed by an appropriate projection to our problem definition. After that, the work focuses on the training and explanation of skin lesion image classification algorithms via several state-of-the-art explainable AI methods, some of which are pioneering methods for the given context.

Keywords: Explainable AI, Dermatology, Grad-CAM, LIME, SHAP, Anchors

1. Introduction

Nowadays, thanks to technological advancements as well as ongoing research, many innovative approaches are emerging in artificial intelligence (AI). Especially in recent periods, the results of such approaches tend to be remarkable [1]. As of 2021, there have been around 2012 deep learning methodologies that have achieved superhuman performances, such that they have started to lead the accuracy benchmarks of given tasks. As a natural consequence, several fields, more likely with greater growth potential, are aiming to adopt the current trends of AI [2]. Ranging from banking and retail to medicine and healthcare, in a wide spectrum of fields, there are machine learning (ML) algorithms used for several tasks, and it is not excluded that there is even more room for further saturation of ML with these or other fields [3]. The major issue is, however, that the success of AI comes at the price of increasingly complex inner processes taking place inside the algorithm, which in turn makes explainability power lower. With that in mind, the adoption of AI is not an easy task for the other domains' experts, especially if the cases include some sensitive topics like decision-making processes involving humans [5]. The medical domain is a typical example of a case where ethical concerns make the need for transparency of AI models increasingly important. According to statistics, there is a substantial growth, especially in the number of published eXplainable AI (XAI) research papers regarding medical image tasks (Figure 1). Particularly in dermatology, skin lesion classification and detection stand as a vital task, as there can be deadly consequences, especially if there is melanoma involved. Early detection of melanoma is highly desirable because once it is spread into deeper layers of skin, the treatment becomes much more difficult and might have life-threatening consequences [8]. Some alarming facts are that in the US, for women younger than 49 years, melanoma is the third most likely cancer to develop, after thyroid and breast cancers, whereas for men younger than 49 years, melanoma is the most likely cancer to develop than any other cancer [7]. Again, in the US, it is estimated that around 7650 people will lose their lives because of melanoma in 2022 [7]. Awareness of this and more facts about the issue clearly leads medical experts to look for AI's aid. Yet another point to consider, however, is the transparency of AI. Such transparency can be achieved by the ability to explain or interpret the given AI model. To be able to accurately approach the topic, in the next section we will briefly discuss the core concepts of XAI existing in academia.

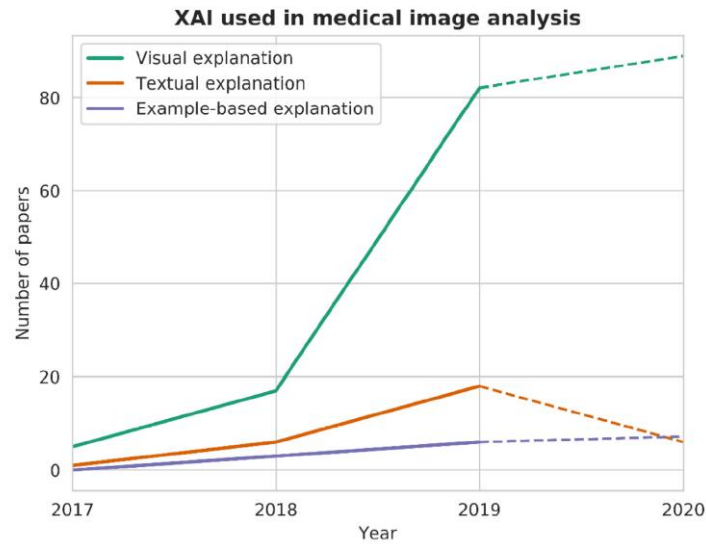


Figure 1. Number of papers published per year in medical image analysis, for the three types of XAI techniques [10]

1.1. Problem Statement

In this work, we are going to apply state-of-the-art XAI techniques to the skin lesion image classification problem. However, in order to be able to accurately approach the task as well as to be clear and justifiable about the further steps taken, it is important to be aware of the existing practices and important concepts related to XAI. Hence, in the next part we will briefly mention the fundamentals and then, in Section 1.3., project the insights back onto our task.

1.2. Fundamentals of XAI – Literature

1.2.1. Interpretability vs. Explainability

First of all, we should be clear about the interpretability and explainability of a model. Often, researchers use these words interchangeably. Although there is no clear separation defined with mathematical rigor, notable efforts have been made to distinguish these two concepts [3] [11]. A popular definition of interpretability by Doshi Velez & Kim (2017) states that "it is the ability to explain or present in humanly understandable terms" [3][9]. Miller (2018), in his definition, describes interpretability as "the degree to which a human can understand the cause of a decision" [5]. Linardatos et al. (2020) in their overview of XAI, after going over major definitions, conclude that interpretability is mostly about the intuition behind the outputs of the model, such that the better the interpretability, the easier it is to identify the cause-effect relationships between the input and outputs [3]. According to Linardatos et al. (2020) and Doshi Velez & Kim (2017), explainability is a tighter concept and it is linked to internal mechanisms of the ML system [3][9]. Thus, making a model explainable means making the inner processes and underlying logic of the algorithm more transparent to humans. Inspired by these, Gilpin et al. (2018) concluded that the explainability of the model is critical for developing transparent methods that can be used in the presence of ethical concerns [12]. Hence, we focus on the term of explainability throughout the work.

1.2.2. Core Concepts

There are several important notions worth considering before moving further with the actual XAI methods. Depending on the problem definition, there might be a number of categorizations of XAI approaches. As mentioned above, the innate nature of XAI is to try to explain AI models—which are usually classified as i) white-box, sometimes called glass-box, where it is possible to intuitively reason about the inner processes, and ii) black-box approaches, whose name speaks for itself—little to no clue for explanation. This work primarily focuses on the latter version. Moreover, according to the technical survey by Das et al., one might differentiate XAI methods by:

a) Scope of explanation:

An explanation can be acquired locally or globally. A local explanation of the model responds to the question, "How does the model behave in a specific regime (location) of a multidimensional space?" In our case, if the doctor looks at a specific skin-lesion image and asks, "Why did AI predict X for this image?", then this question is effectively equivalent to asking for a local explanation around that instance (single skin lesion image) [13]. See Figure 2 for a high-level illustration of locally explainable models. Basically, given a black-box model f , we are trying to come up with a (simpler) model g that explains why the single instance x has the prediction y . The research on local explanation has been founded around the generation of rule-based methods, heatmaps, feature importance matrices, and Bayesian techniques [13]. Nowadays, novel directions in academia aim to boost the performance as well as quality of existing methods by using new approaches such as using attribution maps or game-theory-based models [13]. Activation Maximization [14], Saliency Maps [15], LIME [16], SHAP [17], and Anchors [18] are some of the successful works to note, with the last three being among the most popular methods [4].

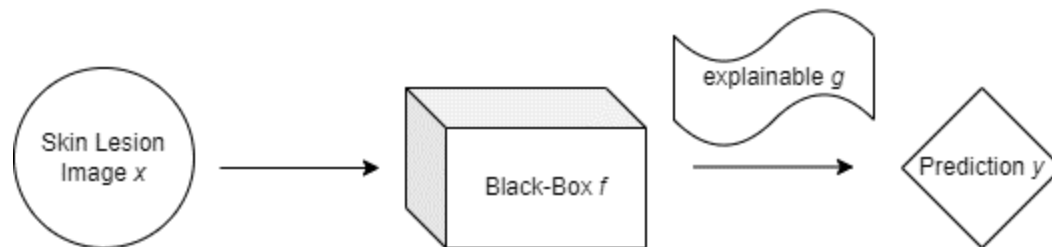


Figure 2. A high-level diagram of locally explainable models

Global explanations, unlike linear explanations, strive to explain the model's behavior not only for a single instance but for the whole domain. Linear models, decision trees, or other rule-based models are examples of inherently globally explainable models. Thus, in essence, global explanation methods are trying to dwindle the black-box models to a globally explainable match accurately enough. The doctor's question in that case would be more like "What is AI thinking about this dataset in general?" See Figure 3. for a high-level illustration of globally explainable models.

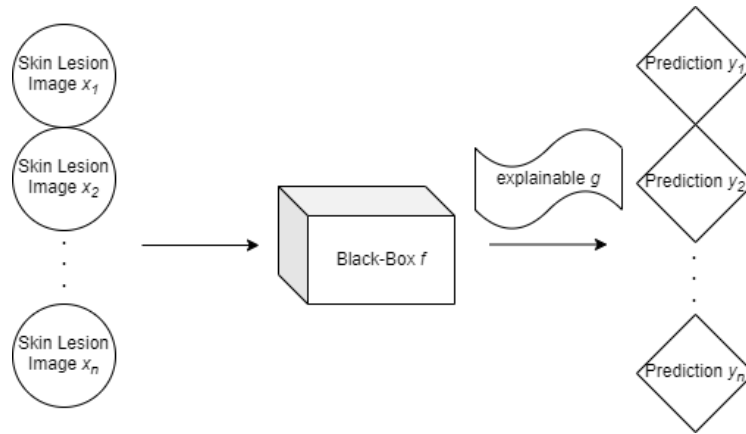


Figure 3. A high-level illustration of globally explainable models.

Given n number of input skin lesion images, an explainable g yields the feature attributions of the given n images – by that the behavior of black-box f on a larger set as well as unseen data is expected to be approximated. Just some of successful works to note down are Concept Activation Vectors [19], Spectral Relevance Analysis [20], Neural Additive Models [21], and Global Surrogate Models (in high level LIME and SHAP can be considered as such [13] – they were also mentioned before for local methods and SHAP is also one of the popular global methods [4])

b) Difference in methodology:

Methodologies of explanation can be classified as perturbation based and gradient based. In general, perturbation based XAI approaches produce explanations by repeatedly using an already trained model with various input changes [13]. LIME, SHAP, and Anchors [18] (a powerful extension of LIME) fall into this category. Gradient-based approaches, on the other hand, use the information flow in a neural network's backward pass to comprehend the neuronal effect and association of the input x to the output. Most gradient-based techniques concentrate on either visualizing highly influential individual neuron activations or general feature attributions adjusted to the input dimensions [13]. Saliency Maps, CAM [22], Grad-CAM [23], and Grad-CAM++ [24] are the notable examples of gradient-based methods.

c) Model Usage or Implementation Level

Methods can be classified as model-intrinsic or post-hoc. Model intrinsic methods can be applied to models that are inherently interpretable, such as rule-based and tree-based models. Hence, naturally, those methods are regarded as model-specific (applicable to only a certain group of models). In contrast, post-hoc algorithms are inherently model agnostic (applicable to any model).

See Figure 4 for the summary diagram of XAI methods. Note the AI box there. That box is essential as it refers to all preliminary actual model training phase before even considering the XAI part – which we have also dealt with in our work prior to XAI (see section 2.2).

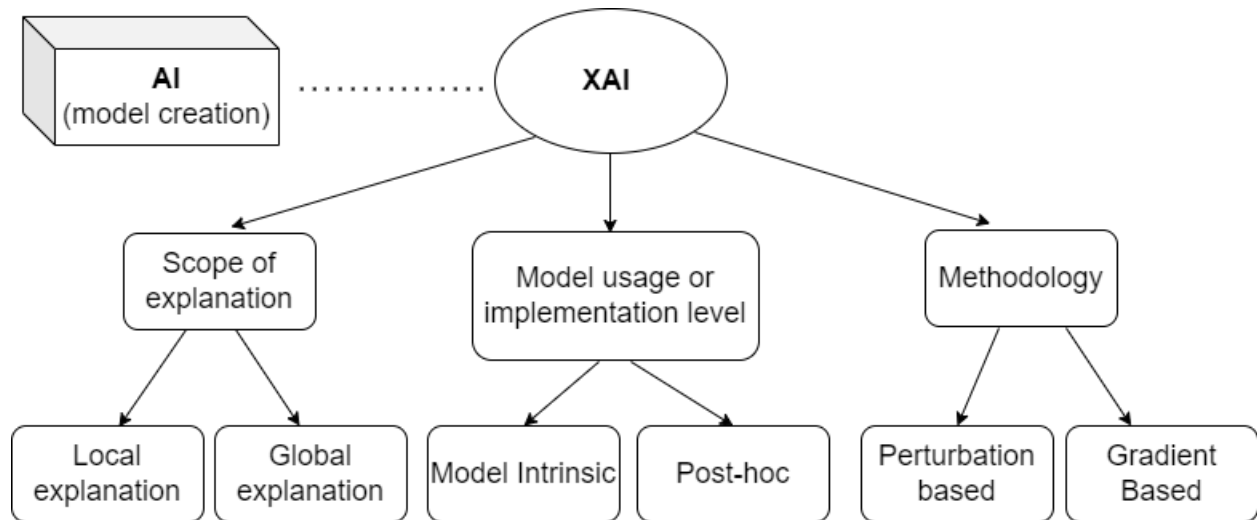


Figure 4. Summary diagram of XAI methods

1.2.3. Evaluation methods of XAI

Besides the awareness of existing methods and their blind application, it is also of the vital importance to know what to do further with the acquired information, so that the explanation can actually be useful for the community. Doshi Velez and Kim (2017) offered three major categories for evaluating XAI outcomes [9]:

- a) Application grounded:
Tries to answer how the acquired results would relate to the domain experts in terms of solving specific tasks.
- b) Human grounded:
Differs from the Application grounded evaluation in a way that the quality checkers are not necessarily domain experts but can be anyone, also the quality of the outcome is defined by how well the general notions are captured, rather than the specific task.
- c) Functionally grounded:
Other than the prior two, human involvement is not needed for this type of evaluation. Rather, there are mathematically rigorous and formally defined criteria (most likely created by applying the prior two methods beforehand) for evaluating the outcomes of explainability.

1.3. Refined Problem View and Our Approach

Now that we have a clearer view of the context, we can move on to introducing our approach. Following the literature review in section 1.2.1, we can justifiably proceed by focusing solely on explainability. Following the section 1.2.2., we attack our task of "Explainable AI in skin lesions" by attempting to cover the diagram of XAI methodologies (in Figure. 4) comprehensively yet effectively. Regarding the leftmost branch of Figure 4. (scope of explanation), we are applying both local (LIME, SHAP, Grad-CAM, Grad-CAM++, Anchors) and global (SHAP) explanation techniques. Regarding the rightmost branch (methodology), we apply both perturbation-based (LIME, SHAP, Anchors) and gradient-based (Grad-CAM, Grad-CAM++) explanation techniques. Lastly, for the mid-branch (model-usage or

implementation level), we are using post-hoc techniques. We are not using model-intrinsic techniques because the problem at hand (skin lesion classification) needs a complex enough model for powerful learning. However, this is a contradiction for model-intrinsic techniques, which, by definition, require the model to be relatively simpler enough to be inherently explainable. See Figure 5. for viewing the updated diagram where green shaded regions imply the directions that we have attacked. Regarding section 1.2.3., we are aiming to use the first two of the explainability evaluation methods. We use application-grounded evaluation by presenting our results to doctors, while in this case, the definition of human-grounded evaluation fits more for presenting the solely XAI-related results to professors. Furthermore, our choice of testing LIME & SHAP traces back to our observation of conflicting opinions in academia about their usage for the skin lesion classification. Metta et al. (2021), for example, consider LIME and SHAP to be two of the major XAI techniques, but they argue that the inner segmentation process of these two methods makes them unsuitable for medical contexts [4]. On the other hand, Hurtado et al. (2022) support that XAI methods, specifically LIME and SHAP, ensure advantages in model result interpretation in melanoma image classification [25]. The paper also adds that LIME performs better than SHAP by means of reproducibility and execution time [25]. On top of that, Anchor is yet another method by the authors of LIME, which offers a number of advantages over LIME [18][26]. To the best of our knowledge, we utilize the Anchors method for the task of skin lesion classification for the first time. Similarly, to the best of our knowledge, we are testing Grad-CAM and comparing it with Grad-CAM++ results for the task of skin lesion classification for the first time. All these processes are follow-ups of a model learning procedure. In Section 2, we discuss in detail our methodologies for that as well.

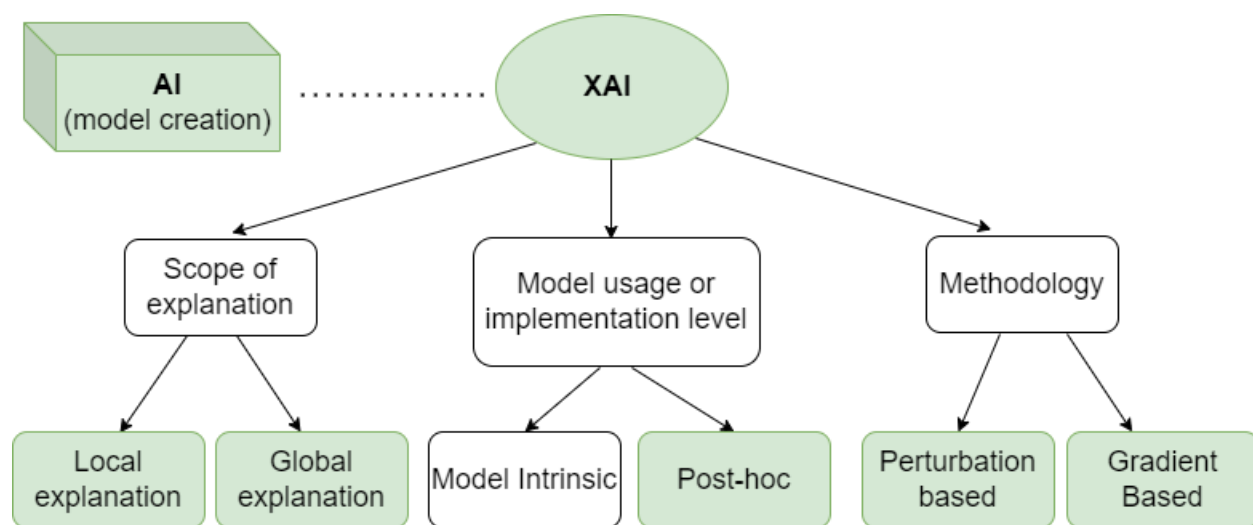


Figure 5. The updated diagram. Green shaded regions imply the directions which we have attacked.

Hence, the contribution of this paper is: A comparative analysis of popular state-of-the-art XAI techniques and the utilization of Grad-CAM++ as well as Anchor for the first time, given a skin lesion classification task.

2. Methods

2.1. Data and Model Set-up

For training and testing purposes, Seven-Point-Checklist [27] and ISIC-2019 public datasets were used. See Figure 6. for the details of Seven-Point-Checklist dataset. There are 5 distinct classes (BCC, NEV, MEL, MISC, SK) that are used as labels in the modelling. As can be seen, the dataset is imbalanced, and this issue was cured via class weighting.

abbrev.	name	7pt-score	# imgs
DIAGNOSIS (DIAG)			
BCC	basal cell carcinoma	-	42
NEV	blue nevus	-	28
NEV	clark nevus	-	399
NEV	combined nevus	-	13
NEV	congenital nevus	-	17
NEV	dermal nevus	-	33
NEV	recurrent nevus	-	6
NEV	reed or spitz nevus	-	79
MEL	melanoma	-	1
MEL	melanoma (in situ)	-	64
MEL	melanoma (less than 0.76 mm)	-	102
MEL	melanoma (0.76 to 1.5 mm)	-	53
MEL	melanoma (more than 1.5 mm)	-	28
MEL	melanoma metastasis	-	4
MISC	dermatofibroma	-	20
MISC	lentigo	-	24
MISC	melanosis	-	16
MISC	miscellaneous	-	8
MISC	vascular lesion	-	29
SK	seborrheic keratosis	-	45

Figure 6. Seven-Point-Checklist dataset information [27]

See Figure 7. for the details of the ISIC-2019 dataset. There are 8 distinct classes (NV, M, BKL, BCC, SCC, VL, DF, AK) that are used as labels in the modelling. Again, the imbalanced data have been cured via class weighting.

Dataset	NV	M	BKL	BCC	SCC	VL	DF	AK	Total
ISIC 2019	12,875	4522	2624	3323	628	253	239	867	25,331

NV—melanocytic nevus, M—melanoma, BKL—benign keratosis, BCC—basal cell carcinoma, SCC—squamous cell carcinoma, VL—vascular lesion, DF—dermatofibroma, AK—actinic keratosis.

Figure 7. ISIC-2019 dataset information

As a machine learning framework, Pytorch was used. Regarding the model learning, transfer learning was done with various versions of ResNet (ResNet34, ResNet50, ResNet101) and VGG (VGG13, VGG16, VGG19).

2.2. Learning

2.2.1. Seven-Point Checklist dataset learning

The dataset has been split into training, validation, and test sets. There have been numerous experiments, including

hyperparameter and network architecture adjustments. Some of the most relevant settings and model performances on the test dataset (for the Seven-Point-Checklist data) are assembled in a database-like fashion in Figure 8.

Models Table:

Model	Batch size	Optimization Algorithm	Learning rate	Dropout rate	Activation function	Epoch number	Performance ID
VGG-16	32	Adam	1e-5	0.5	ReLU	10	#1
VGG-16	32	Adam	1e-5	0.5	ReLU	50	#2
VGG-16	32	Adam	1e-5	0.5	ReLU	350	#3
ResNet50	32	Adam	3e-5	0.3	ReLU	350	#4
ResNet50	32	Adam	3e-5	0.3	Swish	350	#5

Performance Table:

Performance ID	Performance								
#1	Classification report:								
		0	1	2	3	4	accuracy	macro avg	weighted avg
	precision	0.734177	0.500000	0.666667	0.168000	0.096774	0.44557	0.433124	0.619435
	recall	0.529680	0.062500	0.316832	0.525000	0.315789	0.44557	0.349960	0.445570
	f1-score	0.615385	0.111111	0.429530	0.254545	0.148148	0.44557	0.311744	0.488421
	support	219.000000	16.000000	101.000000	40.000000	19.000000	0.44557	395.000000	395.000000
#2	Classification report:								
		0	1	2	3	4	accuracy	macro avg	weighted avg
	precision	0.697778	0.222222	0.525000	0.333333	0.100000	0.531646	0.375667	0.568677
	recall	0.716895	0.250000	0.207921	0.600000	0.210526	0.531646	0.397068	0.531646
	f1-score	0.707207	0.235294	0.297872	0.428571	0.135593	0.531646	0.360908	0.527715
	support	219.000000	16.000000	101.000000	40.000000	19.000000	0.531646	395.000000	395.000000
#3	Classification report:								
		0	1	2	3	4	accuracy	macro avg	weighted avg
	precision	0.827907	0.275862	0.618557	0.7	0.500000	0.711392	0.584465	0.723290
	recall	0.812785	0.500000	0.594059	0.7	0.368421	0.711392	0.595053	0.711392
	f1-score	0.820276	0.355556	0.606061	0.7	0.424242	0.711392	0.581227	0.715449
	support	219.000000	16.000000	101.000000	40.0	19.000000	0.711392	395.000000	395.000000
#4	Classification report:								
		0	1	2	3	4	accuracy	macro avg	weighted avg
	precision	0.792208	0.600000	0.659574	0.571429	0.461538	0.718987	0.616950	0.712245
	recall	0.835616	0.562500	0.613861	0.600000	0.315789	0.718987	0.585553	0.718987
	f1-score	0.813333	0.580645	0.635897	0.585366	0.375000	0.718987	0.598048	0.714369
	support	219.000000	16.000000	101.000000	40.000000	19.000000	0.718987	395.000000	395.000000
#5	Classification report:								
		0	1	2	3	4	accuracy	macro avg	weighted avg
	precision	0.765152	0.350000	0.746667	0.607143	0.625000	0.726582	0.618792	0.720866
	recall	0.922374	0.437500	0.554455	0.425000	0.263158	0.726582	0.520498	0.726582
	f1-score	0.836439	0.388889	0.636364	0.500000	0.370370	0.726582	0.546412	0.710664
	support	219.000000	16.000000	101.000000	40.000000	19.000000	0.726582	395.000000	395.000000

Figure 8. Some settings & model performances on the test dataset (Seven-Point-Checklist data) assembled in a database fashion

It was observed that, for this specific task, Swish activation was doing slightly better than ReLU on average. The model with Performance ID = #5 was used in further steps, e.g. explanation with Grad-CAM(++).

2.2.2. ISIC-2019 dataset learning

Again, the dataset had been split into training, validation, and test sets. Some of the most relevant settings and

model performances on the test dataset (for the ISIC-2019 data) are assembled in a database-like fashion in Figure 9. Figure 10. shows the confusion matrix (of recall metric) for the model with Performance ID = #3. Due to its larger size, the training duration for this dataset was significantly higher than the Seven-Point-Checklist one. Presumably, this fact also manifested itself in the test accuracy, yielding higher values. Lastly, it is important to note that, these models were created so that they could be used in the actual concentration of the paper, which is explainability.

Models Table:

Model	Batch size	Optimization Algorithm	Learning rate	Dropout rate	Activation function	Epoch number	Performance ID
VGG16	32	Adam	3e-5	0.3	Swish	30	#1
ResNet50	32	Adam	3e-5	0.3	Swish	30	#2
ResNet50	32	Adam	3e-5	0.3	Swish	300	#3

Performance Table:

Performance ID	Performance																																																												
#1	<div>Classification report:</div> <table><thead><tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th><th>...</th><th>7</th><th>accuracy</th><th>macro avg</th><th>weighted avg</th></tr></thead><tbody><tr><td>precision</td><td>0.748162</td><td>0.704718</td><td>0.946903</td><td>0.492188</td><td>...</td><td>0.800000</td><td>0.711872</td><td>0.717190</td><td>0.750316</td></tr><tr><td>recall</td><td>0.966034</td><td>0.769103</td><td>0.130329</td><td>0.802548</td><td>...</td><td>0.636364</td><td>0.711872</td><td>0.620504</td><td>0.711872</td></tr><tr><td>f1-score</td><td>0.843253</td><td>0.735504</td><td>0.229122</td><td>0.610169</td><td>...</td><td>0.708861</td><td>0.711872</td><td>0.613983</td><td>0.661357</td></tr><tr><td>support</td><td>2002.000000</td><td>602.000000</td><td>821.000000</td><td>157.000000</td><td>...</td><td>44.000000</td><td>0.711872</td><td>4262.000000</td><td>4262.000000</td></tr></tbody></table>		0	1	2	3	...	7	accuracy	macro avg	weighted avg	precision	0.748162	0.704718	0.946903	0.492188	...	0.800000	0.711872	0.717190	0.750316	recall	0.966034	0.769103	0.130329	0.802548	...	0.636364	0.711872	0.620504	0.711872	f1-score	0.843253	0.735504	0.229122	0.610169	...	0.708861	0.711872	0.613983	0.661357	support	2002.000000	602.000000	821.000000	157.000000	...	44.000000	0.711872	4262.000000	4262.000000										
	0	1	2	3	...	7	accuracy	macro avg	weighted avg																																																				
precision	0.748162	0.704718	0.946903	0.492188	...	0.800000	0.711872	0.717190	0.750316																																																				
recall	0.966034	0.769103	0.130329	0.802548	...	0.636364	0.711872	0.620504	0.711872																																																				
f1-score	0.843253	0.735504	0.229122	0.610169	...	0.708861	0.711872	0.613983	0.661357																																																				
support	2002.000000	602.000000	821.000000	157.000000	...	44.000000	0.711872	4262.000000	4262.000000																																																				
#2	<div>Classification report:</div> <table><thead><tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>...</th><th>6</th><th>7</th><th>accuracy</th><th>macro avg</th><th>weighted avg</th></tr></thead><tbody><tr><td>precision</td><td>0.737581</td><td>0.826374</td><td>0.951872</td><td>0.356164</td><td>0.631579</td><td>...</td><td>0.863636</td><td>0.717391</td><td>0.709291</td><td>0.711950</td><td>0.761565</td></tr><tr><td>recall</td><td>0.971528</td><td>0.624585</td><td>0.216809</td><td>0.828025</td><td>0.099174</td><td>...</td><td>0.826087</td><td>0.750000</td><td>0.709291</td><td>0.622415</td><td>0.709291</td></tr><tr><td>f1-score</td><td>0.838543</td><td>0.711447</td><td>0.353175</td><td>0.498084</td><td>0.171429</td><td>...</td><td>0.844444</td><td>0.733333</td><td>0.709291</td><td>0.598306</td><td>0.672300</td></tr><tr><td>support</td><td>2002.000000</td><td>602.000000</td><td>821.000000</td><td>157.000000</td><td>121.000000</td><td>...</td><td>46.000000</td><td>44.000000</td><td>0.709291</td><td>4262.000000</td><td>4262.000000</td></tr></tbody></table>		0	1	2	3	4	...	6	7	accuracy	macro avg	weighted avg	precision	0.737581	0.826374	0.951872	0.356164	0.631579	...	0.863636	0.717391	0.709291	0.711950	0.761565	recall	0.971528	0.624585	0.216809	0.828025	0.099174	...	0.826087	0.750000	0.709291	0.622415	0.709291	f1-score	0.838543	0.711447	0.353175	0.498084	0.171429	...	0.844444	0.733333	0.709291	0.598306	0.672300	support	2002.000000	602.000000	821.000000	157.000000	121.000000	...	46.000000	44.000000	0.709291	4262.000000	4262.000000
	0	1	2	3	4	...	6	7	accuracy	macro avg	weighted avg																																																		
precision	0.737581	0.826374	0.951872	0.356164	0.631579	...	0.863636	0.717391	0.709291	0.711950	0.761565																																																		
recall	0.971528	0.624585	0.216809	0.828025	0.099174	...	0.826087	0.750000	0.709291	0.622415	0.709291																																																		
f1-score	0.838543	0.711447	0.353175	0.498084	0.171429	...	0.844444	0.733333	0.709291	0.598306	0.672300																																																		
support	2002.000000	602.000000	821.000000	157.000000	121.000000	...	46.000000	44.000000	0.709291	4262.000000	4262.000000																																																		
#3	<div>Classification report:</div> <table><thead><tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>...</th><th>6</th><th>7</th><th>accuracy</th><th>macro avg</th><th>weighted avg</th></tr></thead><tbody><tr><td>precision</td><td>0.859416</td><td>0.881773</td><td>0.894464</td><td>0.759494</td><td>0.796460</td><td>...</td><td>0.934783</td><td>0.863636</td><td>0.855702</td><td>0.847758</td><td>0.857300</td></tr><tr><td>recall</td><td>0.971029</td><td>0.892027</td><td>0.629720</td><td>0.764331</td><td>0.743802</td><td>...</td><td>0.934783</td><td>0.863636</td><td>0.855702</td><td>0.820332</td><td>0.855702</td></tr><tr><td>f1-score</td><td>0.911820</td><td>0.886870</td><td>0.739099</td><td>0.761905</td><td>0.769231</td><td>...</td><td>0.934783</td><td>0.863636</td><td>0.855702</td><td>0.830595</td><td>0.850414</td></tr><tr><td>support</td><td>2002.000000</td><td>602.000000</td><td>821.000000</td><td>157.000000</td><td>121.000000</td><td>...</td><td>46.000000</td><td>44.000000</td><td>0.855702</td><td>4262.000000</td><td>4262.000000</td></tr></tbody></table>		0	1	2	3	4	...	6	7	accuracy	macro avg	weighted avg	precision	0.859416	0.881773	0.894464	0.759494	0.796460	...	0.934783	0.863636	0.855702	0.847758	0.857300	recall	0.971029	0.892027	0.629720	0.764331	0.743802	...	0.934783	0.863636	0.855702	0.820332	0.855702	f1-score	0.911820	0.886870	0.739099	0.761905	0.769231	...	0.934783	0.863636	0.855702	0.830595	0.850414	support	2002.000000	602.000000	821.000000	157.000000	121.000000	...	46.000000	44.000000	0.855702	4262.000000	4262.000000
	0	1	2	3	4	...	6	7	accuracy	macro avg	weighted avg																																																		
precision	0.859416	0.881773	0.894464	0.759494	0.796460	...	0.934783	0.863636	0.855702	0.847758	0.857300																																																		
recall	0.971029	0.892027	0.629720	0.764331	0.743802	...	0.934783	0.863636	0.855702	0.820332	0.855702																																																		
f1-score	0.911820	0.886870	0.739099	0.761905	0.769231	...	0.934783	0.863636	0.855702	0.830595	0.850414																																																		
support	2002.000000	602.000000	821.000000	157.000000	121.000000	...	46.000000	44.000000	0.855702	4262.000000	4262.000000																																																		

Figure 9. Some settings & model performances on the test dataset (ISIC-2019 data) assembled in a database fashion

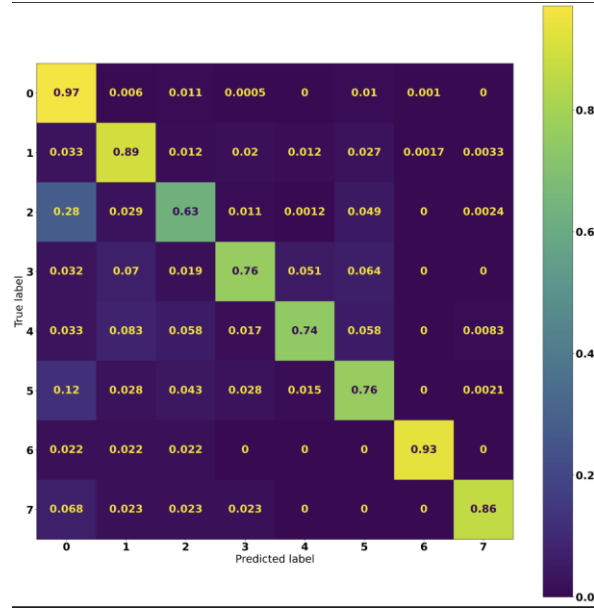


Figure 10. Confusion matrix (of recall metric) for the model with Performance ID = #3

2.3. Explainable AI

In section 1.3, we have mentioned our approach to explainability. Due to the volume of scientific work in the field, Linardatos et. al (2020) have suggested two subcategories of explanation methods:

- a) Explaining deep learning models
- b) Explaining any black-box model

Correspondingly, the way we organize our approach is aligned with this categorization.

2.3.1. Explaining deep learning models

2.3.1.1. Grad-CAM

Grad-CAM (Gradient Class Activation Maps) solves certain limitations of its predecessor CAM (Class Activation Maps) and is indeed a strict generalization of it. Specifically, Grad-CAM can be applied to any convolutional neural network with any shape and can produce visual explanations. As its name hints (as was also mentioned in Section 1.), Grad-CAM is a gradient-based method. Its basic workflow includes utilization of the class-specific gradient information at the last convolutional layer for producing a coarse localization map of important (e.g. most contributing during the classification) areas of the image [3][23]. We are using Grad-CAM for our classification task, but in fact, it can be applied to other types of tasks such as image segmentation and visual question answering. Figure 11. demonstrates the architecture and workflow of Grad-CAM for the classification task of a toy example.

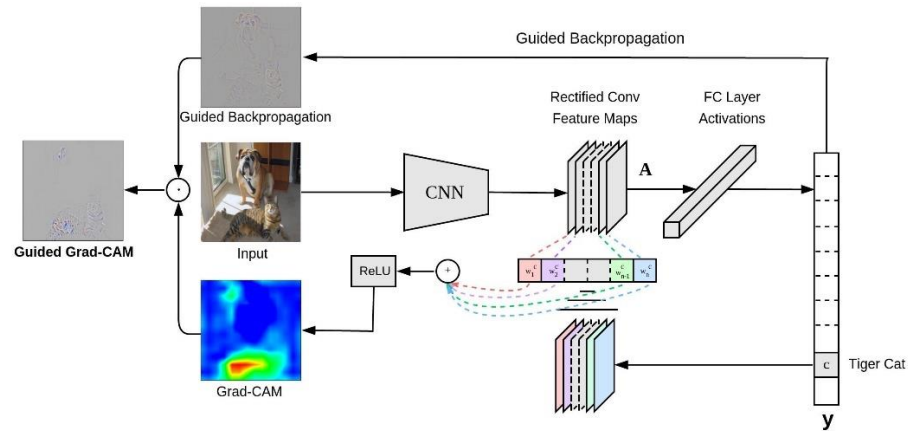

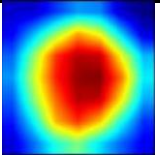
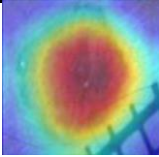

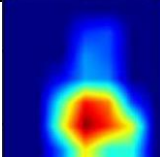
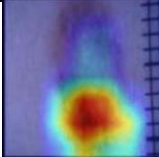
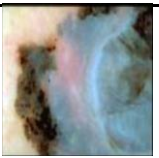
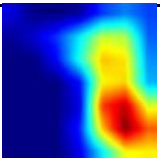
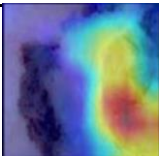

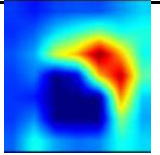
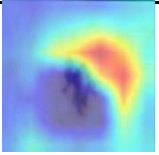


Figure 11. Architecture and workflow of Grad-CAM for the classification task of a toy example

Given an instance's (dog or cat image in the toy example of Figure 11. or a skin lesion image in our case) prediction (husky, tiger cat, etc. in the example or melanoma, nevus, etc. in our case), we ask what parts of the image were positively (and to what extent) contributing to the process of retrieving the current prediction. Grad-CAM returns only positively contributing parts, because of the ReLU in the workflow. Thus, Grad-CAM could be useful to probe whether the highly accurate predictions of powerful models are generated indeed via considering areas of the images that are also reasonable enough for human experts to pay attention to. See Figure 12., for the Grad-CAM results of some test images from the Seven-Point-Checklist and ISIC-2019 datasets, where models with Performance ID = #5 (for Seven-Point-Checklist, see Figure 8.) and with Performance ID = #3 (for ISIC-2019, see Figure 9.) were used.

Image	Grad-CAM	Image + Grad-CAM	Dataset	True Label, Prediction (of respective dataset)
			Seven-Point-Checklist	1,1
			Seven-Point-Checklist	0,0
			Seven-Point-Checklist	2,1
			Seven-Point-Checklist	1,0


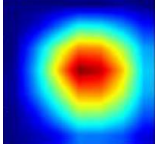
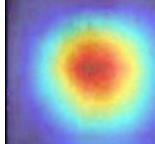

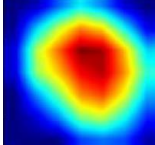
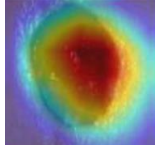
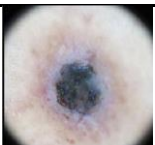
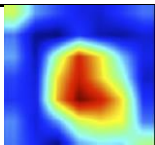
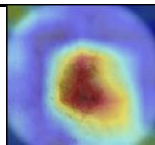

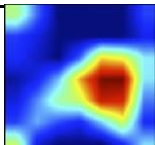
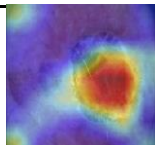
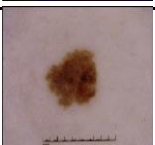
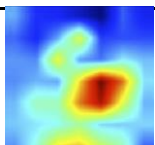
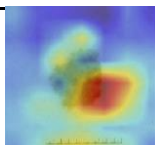

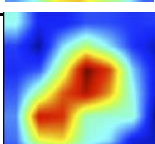
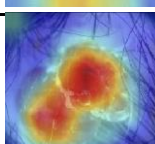
			Seven-Point-Checklist	3,3
			Seven-Point-Checklist	2,2
			ISIC-2019	1,1
			ISIC-2019	2,2
			ISIC-2019	2,0
			ISIC-2019	6,6



Figure 12. Grad-CAM results of some test images from the Seven-Point-Checklist and ISIC-2019 datasets, where models with Performance ID = #5 (for Seven-Point-Checklist, see Figure 8.) and with Performance ID = #3 (for ISIC-2019, see Figure 9.) were used.

There are (at least) two important nuances that are worth noting. First, in most of the correctly classified instances, Grad-CAM visualization intuitively makes sense. Second, in most of the wrongly classified instances, Grad-CAM visualization is far from intuition (for example, look at 4th instance in Figure 12.).

Although it is an upgrade from CAM, Grad-CAM also has its limitations. One limitation is that it cannot localize multiple occurrences of an object within an image. Also, because of its partial derivative assumptions, Grad-CAM is not able to accurately point out the class regions in a given image [3]. Grad-CAM++, an extension of Grad-CAM, comes with a promise by its authors that it is able to furnish better visual explanations. In the next part, we will be discussing how Grad-CAM++ looks in our task.

2.3.1.2. Grad-CAM++

In Grad-CAM++, to put it more precisely, object localization is expanded to include multiple object instances in a single image. Here, a weighted combination of the positive partial derivatives of the last convolutional layer feature maps with respect to a particular class score is used as weights in order to produce a visual explanation for the respective class label. Figure 13. demonstrates the structures of all three variations of CAM methods (CAM, Grad-CAM, Grad-CAM++).

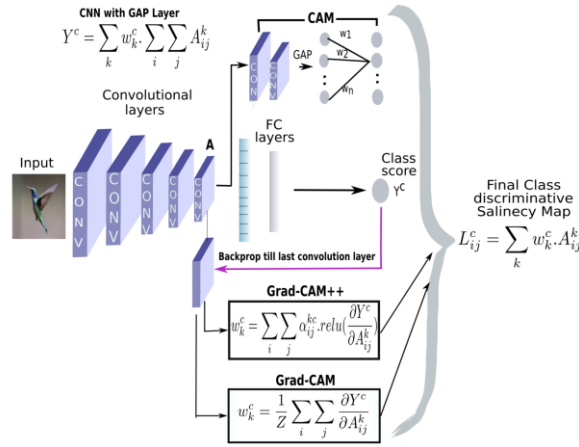


Figure 13. Overview of all the CAM methods

Considering the upgrade from Grad-CAM, Grad-CAM++ is expected to be especially powerful if the task is multilabel classification within a single instance – which is not the direct aim of our task. However, authors of Grad-CAM++ also show that, even if the objective is the classification of a single label within a given instance (it is possible to have multiple occurrences of the same class instance within an image), Grad-CAM fails to accurately localize the areas of multiple same-class occurrences within the image, which might be a disadvantage for the case of skin lesions. Moreover, as can be seen from Figure 13. Formulas in squared boxes, in Grad-CAM, the weights for particular feature map and class c are calculated as following:

$$w_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial Y^c}{\partial A_{ij}^k}$$

Here Z denotes number of pixels in the activation map, A^k is k^{th} activation map, Y^c is a differentiable function of A^k . However, by this approach the w_k^c 's are calculated in a plain way (unweighted) of averaging of partial derivatives over the number of pixels (Z). This results in the localizations where not the whole object is captured but only some bits of it – which in turn may damp the explanation. To address this issue, Grad-CAM++ suggests another way of calculating w_k^c :

$$w_k^c = \sum_i \sum_j \alpha_{ij}^{kc} \cdot \max\left(0, \frac{\partial Y^c}{\partial A_{ij}^k}\right)$$

Here α_{ij}^{kc} are the pixel-coefficients for pixel-wise gradients for class c and feature map A^k . Note that the Grad-CAM is the case where $\alpha_{ij}^{kc} = 1/Z$.

Seeking for updates in the explanations, we present the comparisons between Grad-CAM vs Grad-CAM++ is Figure 14.

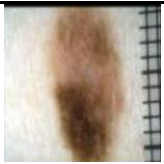
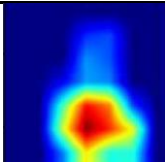
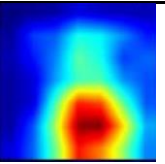
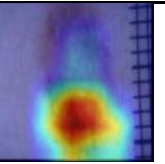
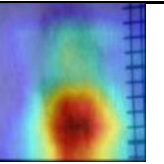

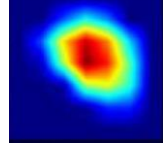
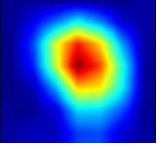
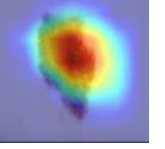
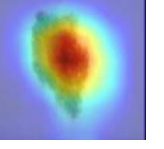

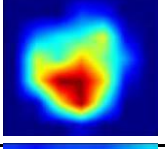
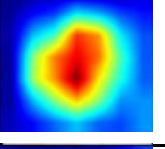
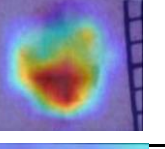
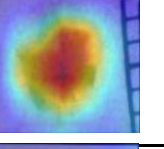

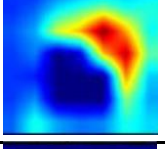
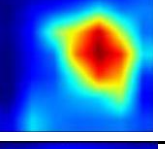
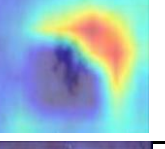
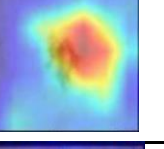

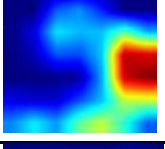
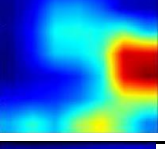
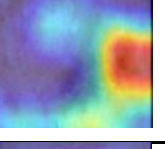
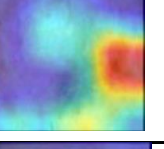

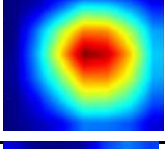
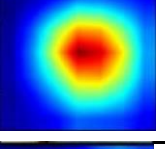
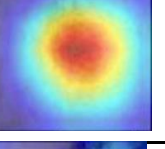
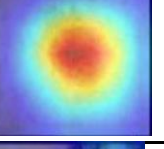

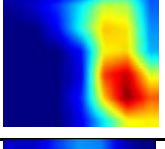
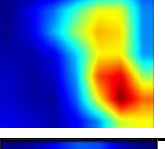
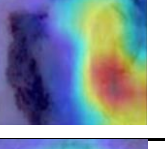
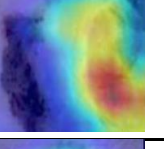

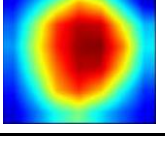
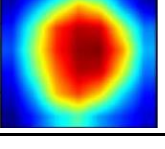
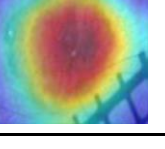
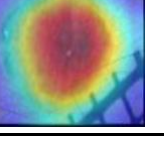
Image Instance	Grad-CAM	Grad-CAM++	Image + Grad-CAM	Image + Grad-CAM++	True Label, Prediction (of respective dataset)
					0,0
					0,0
					0,0
					0,1
					2,2
					3,3
					2,1
					1,1

Figure 14. Comparisons between Grad-CAM vs Grad-CAM++ performances

According to our results, Grad-CAM++'s claim of covering the class areas better than Grad-CAM indeed holds for some instances (1st, 2nd, 3rd, 4th, slightly 5th row in Figure 14.). Yet, for some other instances, the explanations are

almost identical to those of Grad-CAM. But this comes as no surprise, as the authors of Grad-CAM++ have mentioned that in their tests and experiments, in a little more than 1/3rd of the cases, they were getting similar Grad-CAM and Grad-CAM++ results (as Grad-CAM++ can be considered as a generalized version of Grad-CAM). Our main observation is that when the instance is correctly classified, Grad-CAM++ tends to capture more of the lesion zone or captures like Grad-CAM, but not worse. Even in wrongly classified cases, we can see Grad-CAM++ is trying to localize the lesion zone better than Grad-CAM (e.g., 4th row in Figure 14.). We conclude our view of Grad-CAM++ by noting that it promises better potential for its already well-performing predecessor, Grad-CAM.

2.3.2. Explaining any black-box model

After considering the part with the methods for explaining deep learning models, we now move on with explanation methods for any black-box model.

2.3.2.1. *LIME*

LIME (Local Interpretable Model-agnostic Explanations) is one of the most popular methods for explainability that is applied to black-box models. LIME can give explanations for a single prediction score produced by any classifier using a straightforward yet effective methodology. Simulated randomly sampled data is generated in the vicinity of the input instance for which a specified instance and its accompanying prediction are made. Using the black-box model, new predictions are then computed for each of these randomly generated instances and weighted as to how close to the input instance their corresponding instances are. Finally, this newly generated dataset of perturbed examples is used to train a simplistic, understandable model, such as a decision tree. The initial black box model is ultimately interpreted by interpreting this local model [5]. Figure 15. shows a simple example just to demonstrate how the LIME works. Given an instance, the red colored bold plus shape in the image, a certain number of points are generated around its neighborhood and then predicted by the black box. The closer the prediction points to the bold plus shape, the bigger their sizes are. Then a simpler model, in this case linear, is fit, hinting at the behavior of the blackbox in that particular area, which is effectively a local explanation. Figure 16. shows our findings upon application LIME on our context. The yellow-drawn areas locate superpixels that contributed most to the prediction of the respective label.

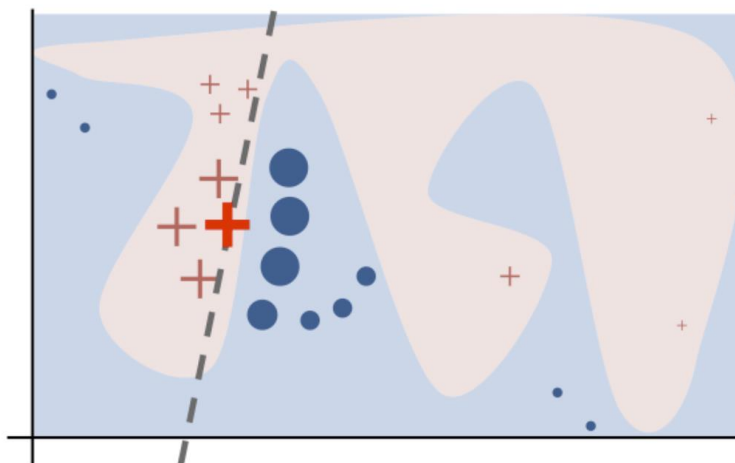


Figure 15. A simple example just to demonstrate how the LIME works

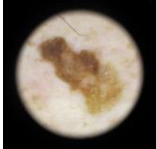
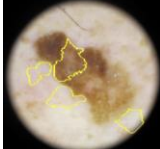
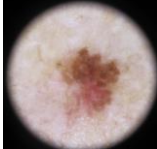




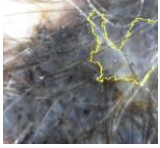
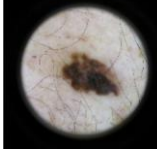

Image Instance	LIME explanation	Number of generated samples	Number of features assumed
		5000	3
		5000	3
		1000	3
		1000	3
		1000	3

Figure 16. Performance of LIME

Looking at the results, we can see that while in some cases explanations are intuitive, for other cases they are not. This is very likely due to the stochastic nature of the algorithm. Indeed, Garreau et al. (2020) in the first ever theoretical analysis of LIME, had approved the significance and meaningfulness of LIME, but they had also proved that the inapt parameter choices might lead LIME to miss out important features [5]. Furthermore, Zafar et al. (2019), while introducing deterministic version of LIME, DLIME, had also mentioned that LIME's random perturbation and feature selection procedures result in unstable generated interpretations [5]. According to their report, such unstableness roots from the fact that, for the same prediction different interpretations can be generated (because of a random nature), which can be problematic for industrial use [5]. The authors of LIME later presented another method, Anchor, which we will analyze in section 2.3.2.3. But before that, in the next part we will be viewing yet another popular XAI approach, SHAP.

2.3.2.2. SHAP

Lundberg et al. (2017) have proposed a game-theory inspired explainability method called SHapley Additive

exPlanations (SHAP) [23]. In game theory, Shapley value is a notion for solving problems that entails equally allocating benefits and costs among a group of cooperating actors. Molnar (2022) mentioned another equivalent definition for Shapley values, saying that the Shapley value is the averaged outcome of all the marginal contributions to all possible combinations of the coalitions. Coming to AI domain, the concept of *group of cooperating actors* maps to the *features of the data* at hand. For our case (computer vision), the group of cooperating actors is actually the *(super)pixels* forming up the image. Another way to interpret the Shapley values is to say each feature is a player in the whole team (data), and the prediction is the reward. Then, one follows Shapley values to fairly distribute the reward among the features, where a player can be a single feature (in tabular data) or a group of features (superpixels in our case) [26]. One of the novelties of SHAP is the linear model (additive feature attribution) representation of Shapley value explanations. With this perspective, SHAP is binding LIME and the notion of Shapley values together. SHAP then determine the explanation as following:

$$g(z') = \phi_0 + \sum_{j=1}^M \phi_j z'_j,$$

where g is the explanation of model f , M is the maximum coalition size, and $z' \in \{0,1\}^M$ is the coalition vector.

Figure 15. shows our results of implementing the SHAP explanation of skin lesion classifier (for the ISIC-2019 dataset). VGG16 model was used for this task (more specifically, Performance ID = #1 at Figure 9.). Blue dots represent negative Shapley values (those pixels were negatively contributing to the corresponding label's score function), while red dots represent positive Shapley values (those pixels were positively contributing to the corresponding label's score function). We are using the DeepSHAP extension of SHAP, which is more suitable for our case, as we have a deep neural network hand. DeepSHAP adapts DeepLIFT[29] to supplement the process to be more convenient for compositional properties of deep neural networks, which in turn improves the attributions. In this example, we are using a corpus of 200 images in the background.

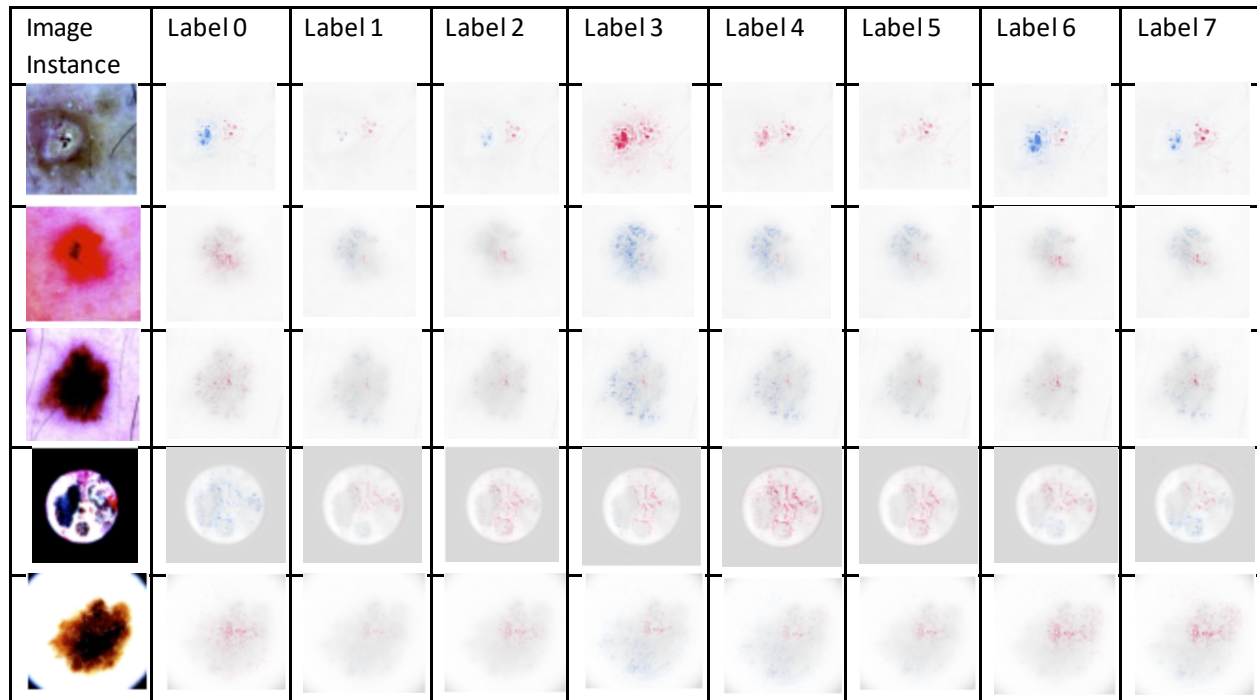


Figure 17. Performance of SHAP

An important note is that, as the features increase, the computational cost increases exponentially (for computing all the coalitions), and takes too many computing hours, which was not the case with the previously mentioned methods (GradCAM, GradCAM++, LIME). It is quite probable that these explanations can be useful for domain experts, who are aware of nuances of the skin lesion regions as well as can interpret the shapley values' meanings. However, from the first glance, the explanations provided by SHAP seem to be relatively more complex than LIME, GradCAM, and GradCAM++. In the next section, we are discussing yet another promising method, Anchors.

2.3.2.3. Anchors

Anchors is another model-agnostic explainability approach, where a high probability of the explanation being true is guaranteed [5] [18]. The anchors approach locates a decision rule that "anchors" the prediction adequately and uses it to explain specific predictions of any black box classification model. If modifications to other feature values have no impact on the forecast, a rule anchors the prediction [26]. Just like its predecessor (LIME), Anchors uses a perturbation-based strategy to produce local explanations for models generated by the black box. The resulting explanations are presented as simple IF-THEN rules, or anchors, as opposed to the surrogate models employed by LIME [26]. To find the anchors, either exploration methods or the multi-armed bandit problem (which has its roots in the field of reinforcement learning) is used. LIME only learns a linear decision boundary that most closely approximates the model given a perturbation space D , hence its findings do not reflect how faithful it is. The anchoring approach, on the other hand, builds explanations whose coverage is tailored to the behavior of the model and the approach clearly states its bounds given the same perturbation space. As a result, Anchor explanations are faithful by design and explicitly define the situations in which they apply. This characteristic makes anchors logical as well as understandable [26]. In the paper, the authors compare both LIME and Anchors, and explain how the neighborhood handling for explanation generation differs between the two algorithms. As a toy example, an illustration was provided by the authors (see Figure 18.), where the model was a black-box binary classifier. If we had had a tabular data, then the algorithm would have created anchors with features, such as: IF SEX = female AND Class = first THEN PREDICT Survived = true WITH PRECISION 97% AND COVERAGE 15% [26] [18]. But in the case of images, the superpixels, which are formed via a segmentation process, are considered features.

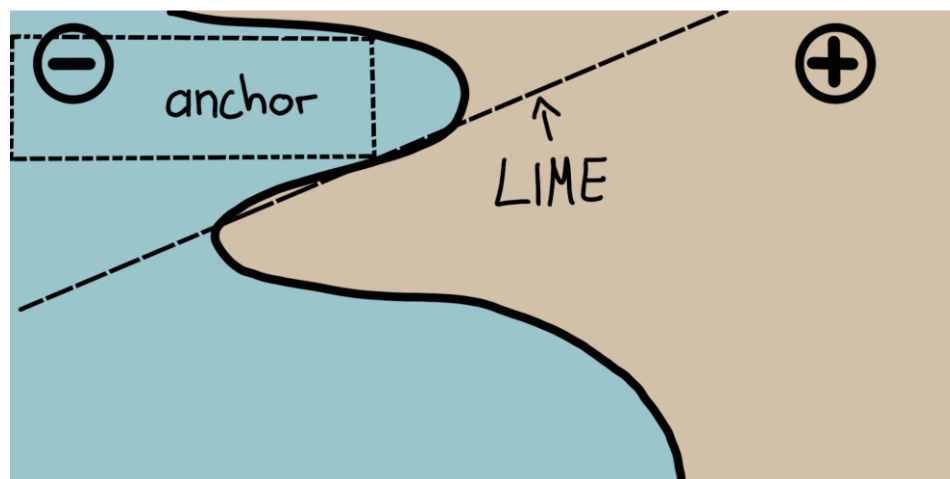


Figure 18. LIME vs. Anchors – A Toy Visualization. Figure from Ribeiro, Singh, and Guestrin (2018)

See Figure 19. for our results with Anchors. The Kmeans algorithm was used for the segmentation procedure of superpixels. The threshold corresponds to the minimum anchor precision threshold value, that is the minimum value

of precision the algorithm tolerates while searching for an anchor that maximizes the coverage. To put it another way, we are looking for an anchor with a precision greater or equal than the given threshold with a confidence of $(1-\delta)$, where we took $\delta = 0.1$. Tau is the multi-armed-bandit parameter used to find anchors at each iteration.

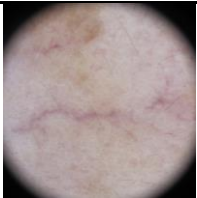


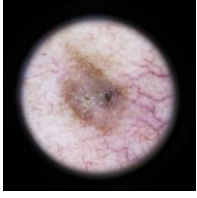

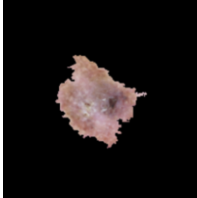
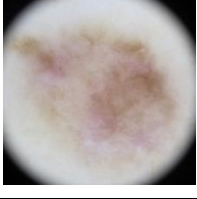








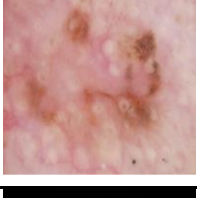

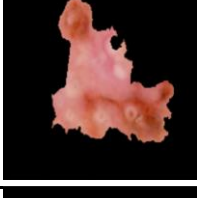
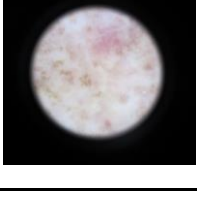


Image Instance	Threshold	Tau	Segmentation (KMeans)	Anchor
	0.75	0.25		
	0.75	0.25		
	0.95	0.15		
	0.95	0.15		
	0.95	0.15		
	0.75	0.25		
	0.75	0.25		

Figure 19. Performance of Anchors

As a result of our implementation of Anchors, the explanations (regions) that anchors provide seem to cover the lesion regions in almost all of the cases and seem to be far from the unintuitive cases we were facing with LIME. The authors of the original paper conducted user research to show that anchors not only result in higher human precision when compared to linear explanations, but also demand less effort on the part of the user in both application and understanding/interpretation. Considering all that, the utilization of Anchors in the domain of skin lesions leaves a promising spot and asks for further research in the area.

3. Conclusion and Future Work

All in all, throughout the paper, we have done an extensive literature review and mapped the common conventions and new trends to our task of skin lesion classification. After the literature review, the overall process can be roughly divided into two parts: learning and explaining. In our work, as part of data, two public skin lesion datasets, Seven-Point-Checklist and ISIC-2019 have been used. Then as part of modelling, we relied on transfer learning of ResNet and VGG models, and we documented the retrieved performances. Afterwards, we divided explanation into two parts: explaining deep learning models and explaining any black-box models. For deep learning model explanations, we have used and compared Grad-CAM++ (for the first time with skin lesions, to the best of our knowledge) and Grad-CAM. According to our experiments, Grad-CAM++ was inclined to cover the class regions more than Grad-CAM. In the next part, we examined LIME, SHAP, and Anchors. Likely to be connected to its stochastic nature, LIME was sometimes generating unintuitive explanations, but was faster than SHAP. SHAP took significantly longer than any other method mentioned, but its explanations may make more sense to domain experts. Anchors were the most promising of the methods among those in Section 2.3.2. Also, due to its scalable and easy-to-use nature, Anchors might outweigh Grad-CAM(++).

It is also important to mention that there were limited time and GPU resources, hence the work could not have been extended for more enhanced results. Undoubtedly, there are many other XAI methods already existing and appearing day-by-day. Other versions of Grad-CAM (XGrad-CAM), LIME (DLIME,QLIME,MPSLIME), and SHAP (KernelSHAP, BSHAP, TreeExplainer) are the variations where even better answers to our questions may be hidden. The incorporation of various contemporary XAI dashboards, such as InterpretML (available at [32]), DrWhy (available at [33]), DeepExplain [34], IML [35], etc., may be another route to take, particularly to make things simpler for non-IT people.

Last but not least, the datasets ISIC-2019 and Seven-Point-Checklist both include meta-data (patient information). Future research may combine such meta-data with image data to retrieve even better models and explanations using the techniques described.

REFERENCES

- [1] LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* 2015, 521, 436–444
- [2] Jordan, M.I.; Mitchell, T.M. Machine learning: Trends, perspectives, and prospects. *Science* 2015, 349, 255–260.
- [3] Linardatos, P., Papastefanopoulos, V., & Kotsiantis, S. (2020). Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy*, 23(1), 18.
- [4] Metta C, Beretta A., Guidotti R. (2021), Explainable Deep Image Classifiers for Skin Lesion Diagnosis
- [5] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *AI*, vol. 267, pp. 1–38, 2019
- [6] Melanoma. The Skin Cancer Foundation. (2022, July 29). Retrieved October 23, 2022, from <https://www.skincancer.org>
- [7] Cancer Facts and Figures 2022. American Cancer Society. <https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2022/2022-cancer-facts-and-figures.pdf> Accessed January 19, 2022.
- [8] Conic RZ, Cabrera CI, Khorana AA, Gastman BR. Determination of the impact of melanoma surgical timing on survival using the National Cancer Database. *J Am Acad Dermatol* 2018; 78(1):40-46.e7. doi:10.1016/j.jaad.2017.08.039.
- [9] Doshi-Velez, F.; Kim, B. Towards a rigorous science of interpretable machine learning. *arXiv* 2017, arXiv:1702.08608
- [10] van der Velden, B. H. M., Kuijf, H. J., Gilhuijs, K. G. A., & Viergever, M. A. (2022). Explainable artificial intelligence (XAI) in deep learning-based medical image analysis. *Medical Image Analysis*, 79, 102470.
- [11] Adadi, A.; Berrada, M. Peeking inside the black-box: A survey on Explainable Artificial Intelligence (XAI). *IEEE Access* 2018, 6, 52138–52160.
- [12] Gilpin, L.H.; Bau, D.; Yuan, B.Z.; Bajwa, A.; Specter, M.; Kagal, L. Explaining explanations: An overview of interpretability of machine learning. In *Proceedings of the 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, Turin, Italy, 1–3 October 2018; pp. 80–89.
- [13] Das, A., & Rad, P. (2020). Opportunities and Challenges in Explainable Artificial Intelligence (XAI): A Survey. *ArXiv*, abs/2006.11371.
- [14] D. Erhan, A. Courville, and Y. Bengio, "Understanding representations learned in deep architectures," Department d'Informatique et Recherche Operationnelle, University of Montreal, QC, Canada, Tech. Rep, vol. 1355, p. 1, 2010.
- [15] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps," 2nd International Conference on Learning Representations, ICLR 2014 - Workshop Track Proceedings, Dec 2013
- [16] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why Should I Trust You?," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*. New York, New York, USA: ACM Press, 2016, pp. 1135–1144
- [17] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, 2017, pp. 4765–4774
- [18] Ribeiro, M. T., Singh, S., & Guestrin, C. (2018). Anchors: High-precision model-agnostic explanations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1). <https://doi.org/10.1609/aaai.v32i1.11491>
- [19] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. Sayres, "Interpretability beyond feature attribution: Quantitative Testing with Concept Activation Vectors (TCAV)," in *35th International Conference on Machine Learning, ICML 2018*, 2018
- [20] S. Lapuschkin, S. Waldchen, A. Binder, G. Montavon, W. Samek, and K.-R. Muller, "Unmasking Clever

Hans predictors and assessing what “ machines really learn,” Nature Communications, vol. 10, no. 1, p. 1096, Dec 2019.

- [21] R. Agarwal, N. Frosst, X. Zhang, R. Caruana, and G. E. Hinton, “Neural additive models: Interpretable machine learning with neural nets,” arXiv preprint arXiv:2004.13912, 2020.
- [22] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, “Learning Deep Features for Discriminative Localization,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, Jun 2016, pp. 2921–2929
- [23] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization,” in Proceedings of the IEEE International Conference on Computer Vision, 2017
- [24] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, “Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks,” in 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, Mar 2018, pp. 839–847.
- [25] Hurtado, S., Nematzadeh, H., García-Nieto, J., Berciano-Guerrero, M.-Á., & Navas-Delgado, I. (2022). On the use of explainable artificial intelligence for the differential diagnosis of pigmented skin lesions. *Bioinformatics and Biomedical Engineering*, 319–329.
- [26] Molnar C. (2022) Interpretable machine learning: A guide for making Black Box models explainable.
- [27] Kawahara, J., Daneshvar, S., Argenziano, G., & Hamarneh, G. (2019). Seven-point checklist and skin lesion classification using multitask multimodal neural nets. *IEEE Journal of Biomedical and Health Informatics*, 23(2), 538–546.
- [28] Springenberg, J.; Dosovitskiy, A.; Brox, T.; Riedmiller, M. Striving for Simplicity: The All Convolutional Net. In Proceedings of the ICLR (Workshop Track), San Diego, CA, USA, 7–9 May 2015.
- [29] M. Sundararajan, A. Taly, and Q. Yan, “Axiomatic attribution for deep networks,” in 34th International Conference on Machine Learning, ICML 2017, 2017
- [30] Garreau, D.; von Luxburg, U. Explaining the Explainer: A First Theoretical Analysis of LIME. In Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, Palermo, Sicily, Italy, 26–28 August 2020; Volume 108, pp. 1287–1296.
- [31] Zafar, M.R.; Khan, N.M. DLIME: A deterministic local interpretable model-agnostic explanations approach for computer-aided diagnosis systems. arXiv 2019, arXiv:1906.10263.
- [32] <https://github.com/interpretml/interpret>
- [33] <https://github.com/ModelOriented/DrWhy>
- [34] M. Ancona, E. Ceolini, C. Oztireli, and M. Gross, “Towards better understanding of gradient-based attribution methods for deep neural networks,” 6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings, 2018.
- [35] C. Molnar, G. Casalicchio, and B. Bischl, “iml: An r package for interpretable machine learning,” *Journal of Open Source Software*, vol. 3, no. 26, p. 786, 2018.