university of groningen / faculty of mathematics and natural sciences / artificial intelligence

Proxy Attention : Comparing and Combining Augmentation with Attention

Graduation Project Proposal
(Computational Intelligence and Robotics)

Subhaditya Mukherjee (s4747925)

Oct 25 2022

Internal Supervisor(s): S.H. Mohades Kasaei, PhD
(Artificial Intelligence, University of Groningen)

Artificial Intelligence
University of Groningen, The Netherlands

1 Introduction

Over the past decade or so, Computer Vision (CV) has taken over the world. Almost every domain, ranging from medicine to robotics has been affected in some way or the other because of it. At the heart of all of these improvements, reside Neural Network architectures. There are thousands of architectures, each made to serve a different purpose, some vastly better than the other. But for each of them, there exists tradeoffs. Accuracy, Memory, Time to train, Cost to run etc. Scaling up these models requires a massive amount of energy, with some consuming upwards of 27,648 kilowatt hours of electricity just to train. This is around the same amount that three households use in a whole year. source. And this is just one model.

In order to get any kind of prediction from them, these networks need to be fed large quantities of data. This training consumes a vast amount of resources that becomes increasingly harder to provide in niche problems where only a small amount of data is available. Models that outperform existing benchmarks such as the Swin Transformer [1] require an abysmal amount of data and energy. This makes it extremely hard for smaller companies, research labs and individuals to use this technology.

The second major flaw of these systems is due to the extremely complex, high dimensional manifolds they try to model. Because of such high dimensionality, it becomes next to impossible to predict exactly why a network made the decision it did. This becomes extremely important in situations like performing medical diagnoses. Not knowing why a network said what it did, makes it very hard to trust. The rise of Explainable AI (XAI) attempts to solve this issue.

In order to tackle the lack of data, many methods such as transforming the present data in multiple ways to increase the available data points aka data augmentation have been created. Methods like transfer learning enable using pre-trained networks to ”fine-tune” on a specific dataset. In the traditional sense, the fields of XAI and data augmentation are not related. So far, the outputs of XAI algorithms have just been used as an explainibility measure, not for training. Therefore, combining these concepts, we arrive at a novel Augmentation technique that uses Saliency maps as an input during training to emulate Attention [2] mechanisms. We call this ”Proxy Attention”.

The objective of this thesis is to design a novel informed augmentation method that would not only reduce the requirement of data, but will also be more memory and time efficient during training as compared to current algorithms in turn building on the advances of XAI to improve training performance.

2 Theoretical Framework

The literature that this thesis builds upon is explained below.

Explainibility

It is commonly noted that neural networks are black boxes. Although in the past decade, these NNs have performed vastly better than a lot of computer vision algorithms, their results are still hard to explain. Due to the high dimensionality of the feature space representation that they create, it becomes impossible to directly visualize the reasons for the decisions made by these networks. The field of Explainable AI (XAI) rose to deal with these challenges. One of the most notable algorithms in this domain for image classification is that of CAM [3] (eventually succeeded by Grad-CAM [4] and then by Grad-CAM++ [5]). The main idea is to visualize the gradients of the final convolutional layer in a trained network to create a sort of ”activation map”. This map could essentially be used to visualize

which parts of the image the network used to make its decision.
In CAM [3], the final layer would have to be replaced by a Global Average Pooling [6] followed by a Softmax. This would then require retraining the network. Since this was not very feasible, further research came up with Grad-CAM [4]. In this paper, the authors proposed using backpropagation over the image and using class information to create the attention map. This makes it able to be directly applied to any network meant for vision classification. The drawback of this method though, was that it sometimes failed when there was more than one instance of the object in the image. It also sometimes failed to fully cover the object in the attention map. Grad-CAM++ [5] solves this issue by modifying the backpropagation algorithm which now scales the map by considering the size of the response.
In this thesis, we want to examine using images weighted by the outputs of Grad-CAM and Grad-CAM++ as augmentation methods during training.

Augmentation

It is not always possible to have a huge amount of data when it comes to training a network. To maximize the performance that can be achieved with existing data, performing transformations to the images either before or during training has become a norm. This is called Augmentation and in a lot of cases, it helps in improving the performance [7]. There are numerous variants of this, ranging from flipping the image on either axis, random cropping, randomly erasing parts of the image, isolating color channels, etc. In this paper, we build on two such techniques.
Random Erasing [8] is an algorithm in a random region from which the image is deleted. The region size is randomly defined at every instance of running the algorithm. Combined with other such Augmentation, this improves performance by a significant amount. The second algorithm is called Visual Context Augmentation [9] in which a network for object detection learns about context by being given data where the object to be detected is blacked out and other images are around it. Many such images with different contexts for the same object are generated and then the network is trained. This teaches the network to predict the object in time.
In this thesis, we combine Augmentation and Explainability to create the proposed ”Proxy Attention”.

Attention

One of the challenges with any neural network is deciding what parts of an image or a sentence are important for the final decision. In Natural Language Processing, Sequence2Sequence models (eg: RNN, LSTM, etc) had been used for a long time, but they had the critical flaw of not being able to work over very long sentences. In time, Vaswani et al. [2], in their seminal paper ”Attention Is All You Need”, proposed a type of attention called Scaled Dot Product Attention as part of a new architecture they titled the ”Transformer”. This modification allowed the network to not only learn how to classify but also learn which parts are important. Transformer based models such as BERT [10] and GPT-3 [11] then suddenly overtook RNNs, LSTMs etc.

ViT

The Transformer used to be generally only used in NLP, but recently [12] converted the Transformer pipeline to work with Images, and the Vision Transformer (ViT) was created. This of course led to a boom in using Transformers for Vision but these networks require even more massive amounts of

data compared to CNNs. Transfer learning alleviates this issue to an extent, but not fully. This is a huge problem.
To examine the effects of Proxy Attention, we compare results to the ViT as well. The aim is to combine them both if this combination ends up being significant.

3 Research Questions

The main research questions that summarize the aims of this study are as follows.

- Is it possible to create an augmentation technique based on Attention maps?
- Is it possible to approximate the effects of Attention from ViTs in a CNN?
- Is it possible to make a network converge faster and consequently require less data using the outputs from XAI techniques?
- Does Proxy Attention impact the explainability of the model in turn?

4 Methods

4.1 Proposed Pipeline

To address the posed research questions, the following pipeline is proposed. Note that these stages might be modified later on as the project progresses and new needs are identified. The objective of reducing memory usage and training time remain the same regardless of the change.
These are shown in Figure 1. This figure is purely for demo and the images shown do not reflect the ones that will be used in the project. The thresholded image was created using Photoshop and is not an output of any kind so far.

Stage 0 - Setup

Since there are a lot of experiments to be run, a configuration file is created with all the parameters required per network and data. Every dataset has it's own path, definition of labels, and image sizes. To ensure compatibility with future research, this type of configuration is very important. Specific Dataloaders for each configuration will have to be created. To ensure proper comparision, the images will all be resized to a common size of 224x224.

Stage 1 - Intial Training

The first stage of the main pipeline is pre-training. This will be done by fine tuning the model on the chosen dataset by transfer learning from a pretrained ImageNet model, if it exists for the chosen model. The number of epochs to train here is yet to be decided. Once this initial training is done, the model is saved for the next stage and the memory is cleared to prevent the GPU from being overloaded. While training, the gradients for the final layer could be saved using Pytorch Hooks which is just a means to ”hook” into the training pipeline and insert arbitrary computation steps.

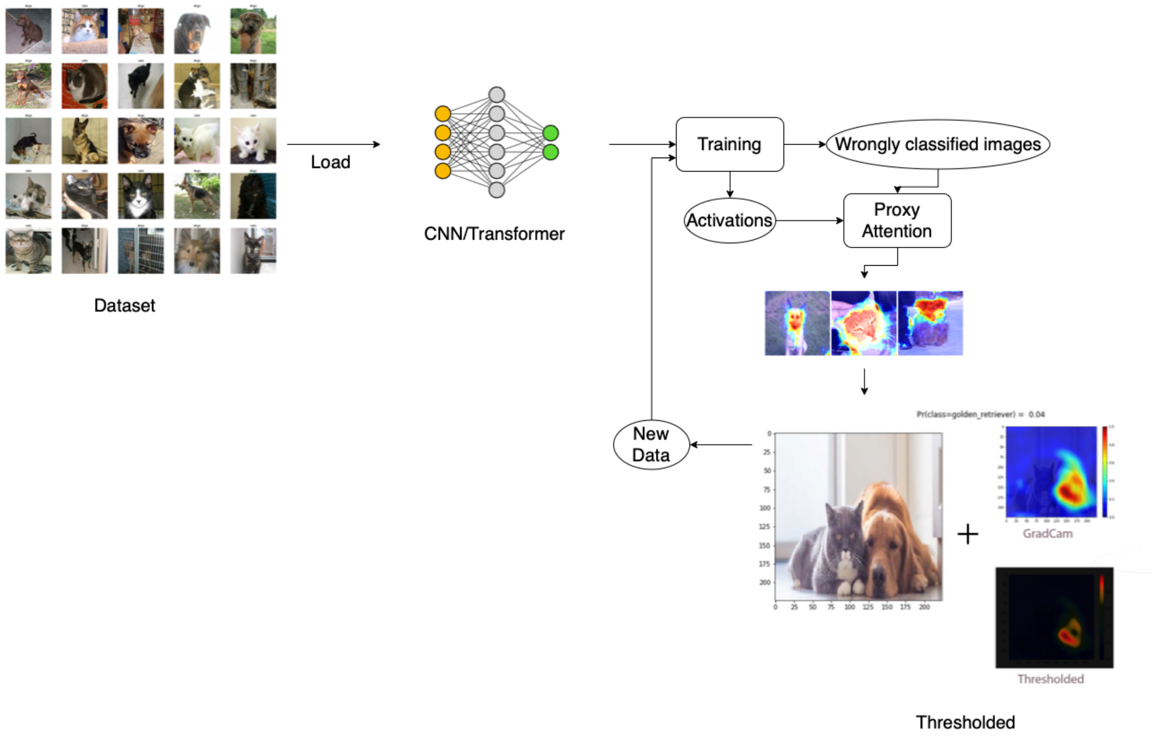

Figure 1: Visualizing the Process

Stage 2 - Proxy Attention

The Proxy Attention stage involves first identifying the images that the model incorrectly classified from the training data. Note that this is not an inference step, but an intermediate, therefore validation data is not used now. After these images are identified, the activations of the final layer wrt each of these images is computed and reshaped into a ”map” with the same size of the image. The highest values of this activation map are thresholded, and these locations are mapped back onto the original image. In the original image, these areas of high activations are replaced with a black pixel. These images are then saved to the disk.

Stage 3 - Retraining

Stage 1 is repeated with the new data along with the original images. Until a satsifactory performance is achieved, this loops back to Stage 1.

4.2 Frameworks

The existing frameworks that will be used to speed up this research effort are as follows:

- Pytorch [13]: For the entire deep learning backbone
- OpenCV [14]: For optimized reading of images
- Fastai [15]: A wrapper on Pytorch that enables faster experimentation

4.3 Dataset and Models

The Datasets and Models to be used are yet to be decided. But in general, the evaluation will start with simpler datasets like the American Sign Language dataset and move onto more complex ones. The same applies to the models. Intial experiments will start with ResNet [16] and move on to more complex models. Eventually the ViT [12] will also be used.

4.4 Evaluation

Evaluation is one of the most important steps to the pipeline in this project. The following metrics are considered now, but research needs to be done on which to include and other metrics to be added.
It is intended that we test around 3 XAI algorithms and 3 datasets. (These have not been researched just yet but will be in the starting weeks of the project)

- The Accuracy, Precision and Recall of the predictions on the validation set
- The time taken to train as compared to the same network without Proxy Attention
- Comparision with ViT [12] and the presence/abscence of Attention
- The amount of GPU RAM used for training

These metrics have to be added into the pipeline to avoid having to manually compute them.

4.5 Tackling Potential Issues

Since this is still an experimental procedure, some issues come to mind. This section aims to provide some answers to them. It is to be noted that further issues may rise during implementation, but these are not known as of now and will be brought to light in time.

Memory Constraints

To reduce memory usage we use the following measures.

1. Unloading the model and it's weights after every stage so it stops using the GPU RAM.
2. Saving the augmented images to the disk instead of keeping them in memory.
3. Using Mixed Precision Training whereby most of the computation is performed in 16bit floats instead of the usual 32bits.
4. Using transfer learning throughout the process.

Computation Time for Activations

Computing the activations can be costly, so some measures will be taken to overcome them.

1. Attempt to parallelize the final prediction over all the images in a batch.
2. Use Pytorch hooks to pre compute these activations as in part they would be computed during training.

Accuracy vs Confidence

While attempting to flesh out this idea, we realized that that it would be interesting to utilize the confidence of predictions along with the accuracy in order to see if this made any difference. This will possibly be discussed in the future.

5 Scientific Relevance for Artificial Intelligence

Training networks on new data is a fundamental task in AI. But for a majority of researchers and startups, the cost of this training is quite high. Also considering the lack of large amounts of data for specific use cases, a need to reduce the training time and require less data during training is ever present.

This research focuses on performing a more informed augmentation that hopes to reduce these problems and allow for faster training with lesser data. In the process, it attempts to use advances in field of XAI to improve training. Further, this project leads into the field of continual learning where adding new information is quite costly and presents a method to reduce this cost.

More specifically, this project aims to make the following contributions to the literature:

- Create a new informed Augmentation method that uses saliency outputs as a sort of Proxy-Attention to improve performance
- Use advances in the field of XAI to empower networks directly
- Contrast using a ViT [12] with a Convnet + Proxy Attention
- Reduce training cost and time of networks so as to aid in continual learning

6 Planning

The following plan contains the work parts (WP) of this graduation thesis and the expected breakdown of how long it would take to work on each of them. There are eight such WPs distributed over 32 weeks as follows :

- WP1 Literature study of gradient based XAI algorithms and overview of project
- WP2 Identifying datasets, metrics, optimizations required to build the pipeline
- WP3 Implementing the base framework, dataset loaders, metrics and optimisations
- WP4 Implementing the gradient based algorithms, Integrating Vision Transformers into the pipeline
- WP5 Perform Training, Experimentation with different modules, Hyper Parameter Optimization
- WP6 Testing and analysis
- WP7 Writing the report
- WP8 Preparing final presentation

The timeline of the research is presented in this Figure 2.

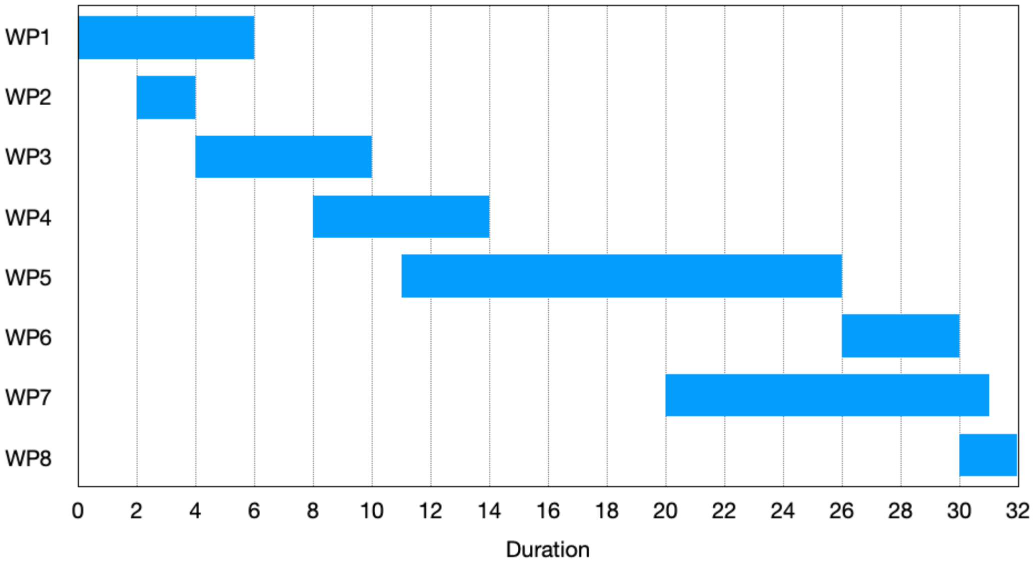

Figure 2: Weekly Plan

7 Resources and Support

For the initial creation, testing and ideation of pipelines and network architecture, I will use my personal computer. It runs Linux and has a decent GPU. For the final testing of multiple architectures and models, I will make use of the Peregrine HPC Cluster's virtual NVIDIA Tesla v-100 nodes so I can run batch experiments.
This work will be directly supervised by Dr. Hamidreza Kasaei. A co-supervisor is yet to be determined. The plan for meetings is to have a progress update every two weeks until completion. This schedule should remain stable for most of the project. There is no collaboration with any company so other related points do not apply here.

References

[1] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, Y. Wei, J. Ning, Y. Cao, Z. Zhang, L. Dong, F. Wei, and B. Guo, "Swin Transformer V2: Scaling Up Capacity and Resolution," Apr. 2022. arXiv:2111.09883 [cs] version: 2.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," Dec. 2017. arXiv:1706.03762 [cs].

[3] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Is object localization for free? - Weakly-supervised learning with convolutional neural networks," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Boston, MA, USA), pp. 685–694, IEEE, June 2015.

[4] 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," p. 9.

[5] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Improved Visual Explanations for Deep Convolutional Networks," in *2018 IEEE Winter Confer-*

ence on Applications of Computer Vision (WACV), pp. 839–847, Mar. 2018. arXiv:1710.11063 [cs].

[6] M. Lin, Q. Chen, and S. Yan, "Network In Network," Mar. 2014. arXiv:1312.4400 [cs] version: 3.

[7] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017.

[8] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, pp. 13001–13008, 2020.

[9] N. Dvornik, J. Mairal, and C. Schmid, "Modeling visual context is key to augmenting object detection datasets," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 364–380, 2018.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," May 2019. arXiv:1810.04805 [cs].

[11] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language Models are Few-Shot Learners," July 2020. arXiv:2005.14165 [cs].

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[13] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[14] G. Bradski, "The opencv library.," *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, vol. 25, no. 11, pp. 120–123, 2000.

[15] J. Howard *et al.*, "fastai." `https://github.com/fastai/fastai`, 2018.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.