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

- Transformers taken the NLP world by storm -¿ CV now
- Possible to use attention methods in CNNs : ConvNet for the Roaring 20s
- ??? Why not use transformers -¿ Memory, Time to train, More data etc
- XAI. Salience maps. GradCAM. Outputs for XAI.
- Attention.
- Papers on Augmentation

2 Theoretical Framework

Explainibility

Augmentation

Attention

ViT

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 using the outputs from XAI techniques?

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.

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.

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[1]: For the entire deep learning backbone
- OpenCV[2]: For optimized reading of images
- Fastai[3]: 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 [4] and move on to more complex models. Eventually the ViT [5] 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.

- 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 [5] 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, I 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 [5] 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 CAM-like 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 CAM-like 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 1.

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.

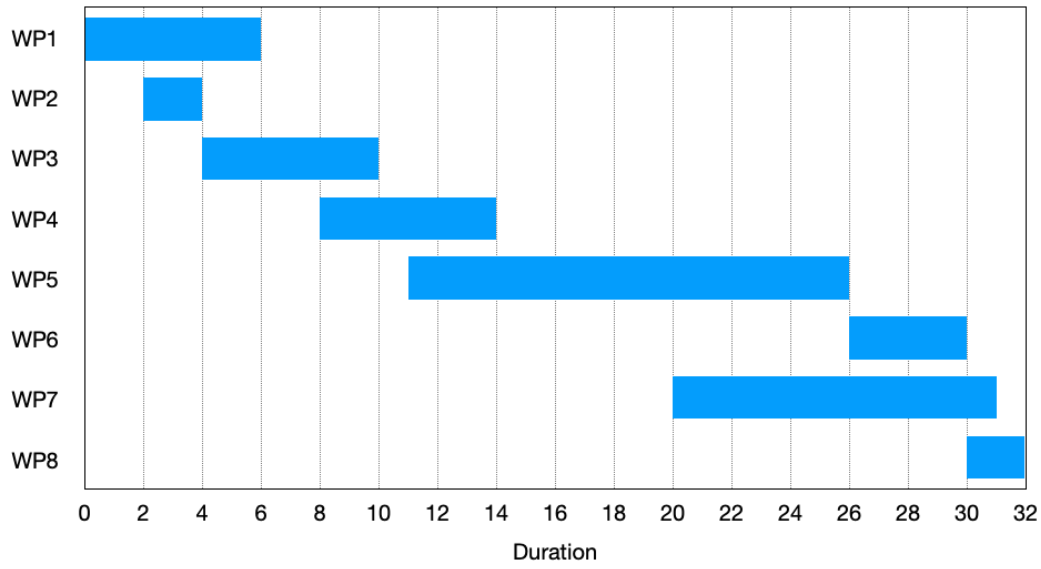

Figure 1: Weekly Plan

References

[1] 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.

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

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

[4] 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.

[5] 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.