M.Sc. Artificial Intelligence (Extended) Research Project Agreement

PERSONAL INFORMATION STUDENT

Family name : Kimman

First name : Nils

Student number : s1007368

E-mail address : nilskimman@xs4all.nl

Course code / ECs : MKI94 (45 EC) (please select one)

Note: unless otherwise decided by the exam board, students that started the programme in the study year 2017-2018 take MKI92 or MKI94; students started before take MKI92 or MKI93.

AI specialisation: (please select one)
(2019-later) Cognitive Computing

Supervision

Project type : internal (please select one)

Note: internal projects are supervised and assessed by AI staff members. Affiliated projects are supervised by an affiliated staff member (e.g., DCC, CS, MPI, CLS etc) and assessed by an AI staff member. External (non-affiliated) internships are supervised by both an external supervisor and an AI staff member (mostly within a thesis circle).

Details of Supervisor 1 (Internal supervisor and/or assessor – must be AI staff member)

Name :Umut Güçlü

Details of Supervisor 2 (Internal / Affiliated / External supervisors)

Name :Thirza Dado?_____

Institute :Radboud/Donders?_____

Telephone :_____

E-mail address :_____

(*Add if there are more supervisors involved*)

Details of hosting institute

Name :Radboud University + Donders

Address :_____

Contact person :_____

Telephone :_____

E-mail address :_____

PROJECT DESCRIPTION

1. TITLE OF PROJECT: USING ENCODING MODEL AFRT TO VISUALISE NEURON RECEPTIVE FIELDS

2. ABSTRACT:

3. PROJECT DESCRIPTION: (MAX. 2000 WORDS) *1559*

BACKGROUND OF PROJECT (THEORY, STATE OF THE ART)

Spatial transformer networks [1] are a type of neural network that, as the name suggests, apply spatial transformations. Specifically, spatial transformers can be used like a module on top of a classical neural network architecture. This allows the neural network to learn the same weights for different orientations of the input data (typically images), thus becoming invariant under spatial transformations. Basically the spatial transformer converts different orientations back to a single canonical form, which causes the network to learn the same weights for images with different orientations. This creates the flexibility and expressiveness needed to model the different spatial orientations.
Although a spatial transformer is not used exactly here, this project does focus on an Affine layer which is heavily based on these ideas. The layer still applies spatial transformations according to affine weights, which are few and very interpretable. For both the x and y axes the affine layer has a scaling, translation, and rotation parameter. This creates sufficient expressiveness to model receptive fields while also maintaining a low dimensionality. This has plentiful advantages, including being less prone to overfitting, more interpretable, and faster to train.

Because of the nature of spatial transformers, these modules are often combined with convolutional neural networks (CNNs) due to their effectiveness in the visual field [6]. CNNs have the ability to deal with the large input size of images without surrendering to overfitting. A traditional neural network would struggle to find the expressiveness without losing almost all generalisability. The convolutional layers that a CNN uses solve this by looking at local neighbourhoods using a kernel, and combining information of one pixel with neighbouring pixels using pooling layers. There are many different architectures for CNNs that all perform up to standard, but for this study AlexNet [3] is chosen. However, the main point of AlexNet is not predicting brain responses, it just extracts features. So we need another layer, a single linear response layer, to go from the feature space to the brain response space. Put all together, this gives us the Affine Features Response Transform (AFRT) model.

The assumption is that this theory is similar to the way humans perceive objects, based on the idea of receptive fields. Since spatially manipulating an object typically has little effect on how we perceive it, an encoding model should have a similar mechanism. In humans, receptive fields make sure that our visual perception is invariant under spatial transformations (to a certain extent). Because different neurons activate for different stimuli, information about the spatial orientation of objects is encapsulated in the brain [4]. So there is a similarity between the biological and the artificial mechanisms, which implies that information can be transferred between them. This can be in either direction of course, we can take inspiration from notions like receptive fields to implement them in an artificial network, and observe the results to possibly learn more about how they work.

Spatial transformers typically consist of three components, although we only use two of them. The first component is immediately the one we exclude: a localisation network. This is any neural network that learns the affine parameters to transform the input sample to a canonical pose. Since the affine parameters are learnable parameters of our entire network, this is unnecessary for us. What we do use is the grid generator, which creates a transformed grid according to the affine parameters. Essentially this downscales the image and indicates the receptive field of the neuron. Actually mapping the image onto the grid is done by a bilinear sampler. Important to note about the sampler is that this operation is (sub)differentiable, and very efficient on a GPU.

As previously mentioned, the feature model we use is Alexnet: a deep convolutional neural network used for image classification. It consists of multiple convolutional layers, some of which are followed by pooling layers. The final two layers are fully connected layers where dropout is applied ($p = 0.5$). The only activation function used is the rectified linear unit, on all the convolutional and fully connected layers. The important part is that this model performs very well, and the pretrained network is available online (saving us a lot of time). We use this model to extract features from the images, and determine the brain response via a single fully connected layer. Note that since this model is pre-trained, AFRT only needs to learn parameters for the affine- and response layer. This is important because it drastically reduces the number of parameters that need to be learned. The affine layer has 6 parameters: translation, scale, and rotation for both the x and y-axis, and the response layer has a different number of parameters based on what layer we want to analyse; this ranges from 65 to 385.

There is a similarity between biological neurons and the artificial neurons found in neural networks. It has been shown that the receptive fields of both gradually increase in complexity for e.g. visual processing [2]. This means we can theoretically analyse the receptive fields of the neurons in an artificial neural network to learn something about the receptive fields of biological neurons. The receptive field of a visual neuron specifies the part of the visual field as well as possible transformations for which that neuron activates. For example, some neurons only look at the bottom left corner of an image, or others only look at straight lines. Since this coincides with the function of our Affine layer, the neurons in this layer (or more specifically the affine parameters in this layer) can provide us with insights about the receptive fields of biological neurons

Aim of the project (research question, motivation, impact, importance)

The first and most important step is successfully training an encoding model using the Affine layer. This improves upon classical encoding models by having fewer learnable parameters and being more biologically plausible. The former results in a more interpretable and generalisable model by reducing the dimensionality and thus requiring less regularisation techniques. However, a problem that is not yet fixed is that these models are voxel specific, meaning a lot of models need to be trained in parallel for it to be feasible.
The biological plausibility is important because successfully training this model may allow us to visualise receptive fields of biological neurons. There is a direct mirroring between deep CNNs and the visual pathways in the brain. This means that once the model is trained on real data, we can visualise the hypothesised receptive fields of real neurons. Resulting from this we may find evidence to enforce or refute certain theories regarding the workings of the visual system in the brain. For example, knowing how

receptive fields work can be helpful in treating visual disorders, creating visual prostheses, or improving stimulation in the visual cortex. However, even if this generalisation proves invalid, just looking at the receptive fields of neurons in AlexNet can possibly improve the architecture of deep CNNs.

This results in the following main research question: *"Is it possible to train AFRT and are the resulting affine weights interpretable?"* Since the weights in AFRT do not always directly impact the eventual response scalar, training may not be as straightforward as one might think. On top of that, while such a network may be trained successfully, there is no guarantee that the learned affine weights say anything about the corresponding biological structure. This is why there is a secondary research question: *"What can we learn about human receptive fields based on the learned affine weights?"*.

PROJECT PLAN (APPROACH, METHODS, DESIGN, ANALYSES)

The basic premise is training the AFRT model parameters for a specific AlexNet layer. The parameters of interest then are of course the affine transformations for specific samples. However, these have to be learned from data that is put through a feature model and a dense layer (which has to be learned synchronously), meaning learning is not straightforward. So it is likely that training is very strict and has to be tested in steps, increasing the complexity gradually. This means: firstly working on simulated data so the ground truths are known, just learning scaling and translation (ignoring rotation), just learning the affine layer and fixing the response layer, and only looking at layer 1 of AlexNet.
Simulating the brain response data is actually very straightforward. This is done by choosing parameters beforehand and saving the model's output, which is the response of a single voxel as a scalar per sample. Even though this fake brain response is likely incorrect or unrealistic, the expressiveness of the model should be such that it can learn these parameters regardless of this phenomenon. Now we can compare the model parameters to the ground truths directly to check if the model learned the correct thing. Note that this would have been impossible with the real data since it is unknown what affine parameters the receptive fields of the biological neurons would have, or the artificial equivalent of them.
The real data used will be [monkey data], which consists of fMRI measurements. The stimuli presented come from the THINGS dataset [5], which is a dataset consisting of many categories of objects. These provide enough visual complexity to distinguish different brain signals from each other, and are even labelled by category. These labels will likely remain unused, but having access to them for testing purposes is a nice benefit.

Training the network consists of testing different hyperparameters, optimizers, and possibly architectures. Since we use simulated data, a loss of 0 is obtainable, and we are only comparing brain responses (scalars), so mean squared error works fine as a loss. Because the THINGS dataset is so large, a subset will be used to speed up testing; however this means that hyperparameters such as the batch size may have different optimal values for the real data and the simulated data. On top of that, since we decide the ground truth values of the affine parameters ourselves, we can experiment with extreme values to see what is still learnable and what is not (and ponder why?).

4. Schedule: (max. 1 page)

Specifically state the start and end dates of the various phases of the project

01/09/2022 - 01/11/2022 - training AFRT on simulation data on a single layer
01/11/2022 - 01/01/2022 - implement real data and train again
01/01/2022 - 01/02/2022 - generalise to different Alexnet layers and further experimentation
01/02/2022 - 01/04/2022 - analyse results and write report

5. Scientific, societal and/or technological relevance: (About 250 words)

Describe here the broader context and relEvance of your project

Alexnet is an often used network for image classification tasks, so learning more about how it works is always beneficial. On top of that, learning the receptive fields in Alexnet can help develop new CNN architectures. These architectures can perhaps achieve better performance in certain situations, or provide further insight into neural networks.
Learning more about biological neuron receptive fields is integral to improving certain visual prostheses. Stimulating the visual area in the case of damaged eyes (the most common visual impairment) can generate phosphenes on the visual field, which can be used to create low resolution binary pictures that detect edges and contours. Knowledge of how visual neurons function can aid this type of research majorly.

6. References:

[1]: Jaderberg, M., Simonyan, K., & Zisserman, A. (2015). Spatial transformer networks. Advances in neural information processing systems, 28.
[2]: Güçlü, U., & van Gerven, M. A. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. Journal of Neuroscience, 35(27), 10005-10014.
[3]: Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25.
[4]: Duhamel, J. R., Bremmer, F., Ben Hamed, S., & Graf, W. (1997). Spatial invariance of visual receptive fields in parietal cortex neurons. Nature, 389(6653), 845-848.
[5]: https://things-initiative.org
[6]: O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458.

7. APPENDIX: INTERNSHIP CONTRACT (if applicable)

aDD HERE ALL FORMAL AGREEMENTS MADE WITH THE host instiute WITH RESPECT TO ALLOWANCES (IF ANY), FACILITIES, CONFIDENTIALLY ETC.

Checks

- € The research proposal was discussed with, and agreed upon by, the master thesis coordinator and the internal supervisor/assessor (AI staff member).

- € At the start of the research project, I (will) have obtained at least 48 EC of course credits in the M.Sc. AI programme.

- € If appropriate, I have discussed the *Checklist for External Projects* with my external and internal supervisor and an internship contract has been agreed upon, checked by the master thesis coordinator, and added as an appendix to the proposal.

Signatures

These signatures confirm the accuracy of all statements made on this form and agree to all principles and articles as stated in the "Rules and Regulations MSc Internship / Research project in Artificial Intelligence" (the most recent document can be found on Brightspace)

Student

Name Date Signature

Internal supervisor (formal assessor)

Name Date Signature

Second internal/affiliated/external supervisor (add more if applicable)

Name Date Signature