Deep learning: proposal final project

Willem Diepeveen, 4391098
Niek van der Laan, 4296915
Daphne van Tetering, 4375165
Paul Verkooijen, 4223322
Kasper Wendel, 4331362

April 28, 2019

1 Introduction

For sequential modelling the canonical method is the RNN. In recent years, other architectures have been proposed and shown to perform better in several cases. One of these approaches is the TCN network, the temporal convolutional network, by Bai et al [1]. This network uses convolutional layers to model the memory of the system. In most of the cases, where a RNN gives good result, the TCN gives equally good or better results. Another, very promising, technique is the Transformer network proposed by Vaswani et al. [9]. This method does not use any recurrences nor convolutions and is very attractive due to its high parallelizability. The network is purely build up from attention mechanisms and fully connected layers. Although these networks have been tested against the accepted RNN variations, these networks are not the standard yet.

In our project we want to look into applying the Transformer network on yet another task, the image to caption task. Xu et al. [11] use a state of the art network consisting of a VGGnet and a LTSM to make an algorithm that turns an image into a caption. We know already that the Transformer can work very well on its own, but we want to know whether this still holds when it gets its input from the VGGnet. We want to investigate whether performance increases as we replace the LSTM with the Transformer network [1]. In short our research question will be:

Does the Transformer network perform better on the image to caption task than the LSTM?

2 Background

Automatically generating a caption for a picture is an task that is very close to the natural understanding that humans have. Current research has shown that this task can be com-

[1] If time permits, we will also look at the TCN.

pleted with the help of Deep Learning in multiple ways [4]. A highly cited paper in this task is from Vinyals et al. [10], that tackled this problem by creating a model that extracts image features with a CNN and uses this information in a LSTM to generate captions. The results of this architecture are promising and a huge improvement of existing methods. An improvement to this model was made by Xu et al. [11] and Jin et al. [6], which added visual attention to it. This was achieved by extracting features from lower convolutional layers and feeding them to the LSTM, instead of combining them with a fully connected layer. This will give the recurrent network more information from different regions in picture, leading to more accurate caption prediction.

In the recurrent networks, different and new architectures are being proposed to solve the problem of sequence modeling. Firstly, the temporal convolutional network (TCN) uses dilations and residual connections with normal convolutions to achieve predictions [1]. Results show that general TCNs outperform LSTMs and RNNs on sequence modeling tasks. Another trend is the transformer model of Vaswani et al. [9] which doesn't use any recurrences nor convolutions but is based on the attention principle. This leads to a speed up as each output of the network does not depend on all previous states like in a LSTM or RNN. Research where a Transformer is used to model captions based on a sequence of image features is currently missing. Therefore, we propose to research this aspect by comparing the performance of the LSTM and the Transformer networks for image caption generating.

3 Data set

The reference paper [11] uses three data sets: Flickr8, Flickr30 and MS COCO. During our project we will first use the Flickr8k data set, since its size allows for a relatively short running time. Afterwards we will use the Flickr30k data set. Our motivations for this are its size and the fact that it is the current standard benchmark. We will not consider the MS COCO data set because it is too big: it takes three days to train on a NVIDIA Titan Black GPU. For completeness, we give a brief overview of each data set below.

Flickr8k

The Flickr8k data set was created by Hodosh et al. [3] and contains 8.092 action images, each with five different captions to describe the depicted entities. Different captions of the same image may focus on different aspects of the scene or use different linguistic constructions. The data set can be obtained from the Department of Computer Science from the University of Illinois [8] and from several GitHub-repositories [2]. An example of an image in the Flickr8k data set is shown in Figure 1 [3].

Flickr30k

The Flickr30k is the current standard benchmark for sentence-based image description. Its structure is similar to the Flickr8k data set discussed previously. The data set contains 31.783 images, each with five captions. The Flickr30k data set was split using a publicly

A man is doing tricks on a bicycle on ramps in front of a crowd.
A man on a bike executes a jump as part of a competition while the crowd watch
A man rides a yellow bike over a ramp while others watch.
Bike rider jumping obstacles.
Bmx biker jumps off of ramp.

Figure 1: An example of an image in the Flickr8 date set with its five captions.

available split [7], since the data set lacked a standardized split. The Flickr30k data set can be obtained from Kaggle [5].

MS COCO

The Microsoft COCO (MS COCO) data set was presented in 2014 and contains 328k images, each with at least five captions per image. In the reference paper, if an image contained more than five images the remainder of the captions was discarded to keep the number of captions consistent throughout the research. To this data set pre-tokenization was applied to keep the tokenization consistent with the one present in the Flickr8k and Flickr30k data sets. The MS COCO data set was also split using a publicly available split [7], because this data set lacks a standardized split as well.

4 Method

4.1 Approach

Similarly to Xu et al. [11] we use a Convolutional Neural Network in order to extract a set of feature vectors which we refer to as annotation vectors. Notably, unlike previous work, Xu et al. choose to extract features from a lower convolutional layer instead of a fully connected layer, allowing the decoder to selectively focus on certain parts of an image by selecting a subset of all feature vectors. However, instead of providing these feature vectors to an RNN, we provide these as an input to a Transformer network, combined with the positional embedded output. To which layer exactly we will input the decoded feature vectors to the Transformer network is still to be investigated. Training of the Transformer network is done with the multi-level supervision training method proposed by Vaswani et al. [9]. Regularization methods to fight overfitting like dropout will be applied where beneficial.

5 Expected results

We want to evaluate the final model with a comparison to Vinyals et al. [10]. Vinyals et al. used three different training sets, Flickr8k, Flickr30k and MS COCO. The Flickr30k uses splits [2]. Our model will be trained using the same splits as the Flickr30k in Vinyals et al..

[2]https://cs.stanford.edu/people/karpathy/deepimagesent/

After training, the testing split will be used and captions will be generated for images. For each caption, a BLUE score will be calculated. The precise brevity penalty for the BLEU metric will be determined later. Vinyals et al. used four different penalties, ranging from 1 to 4. Then the generated BLEU score will be compared with the BLEU scores presented in Vinyals et al.. This comparison will be used to evaluate the usage of a transformer in our model.

6 Feasibility

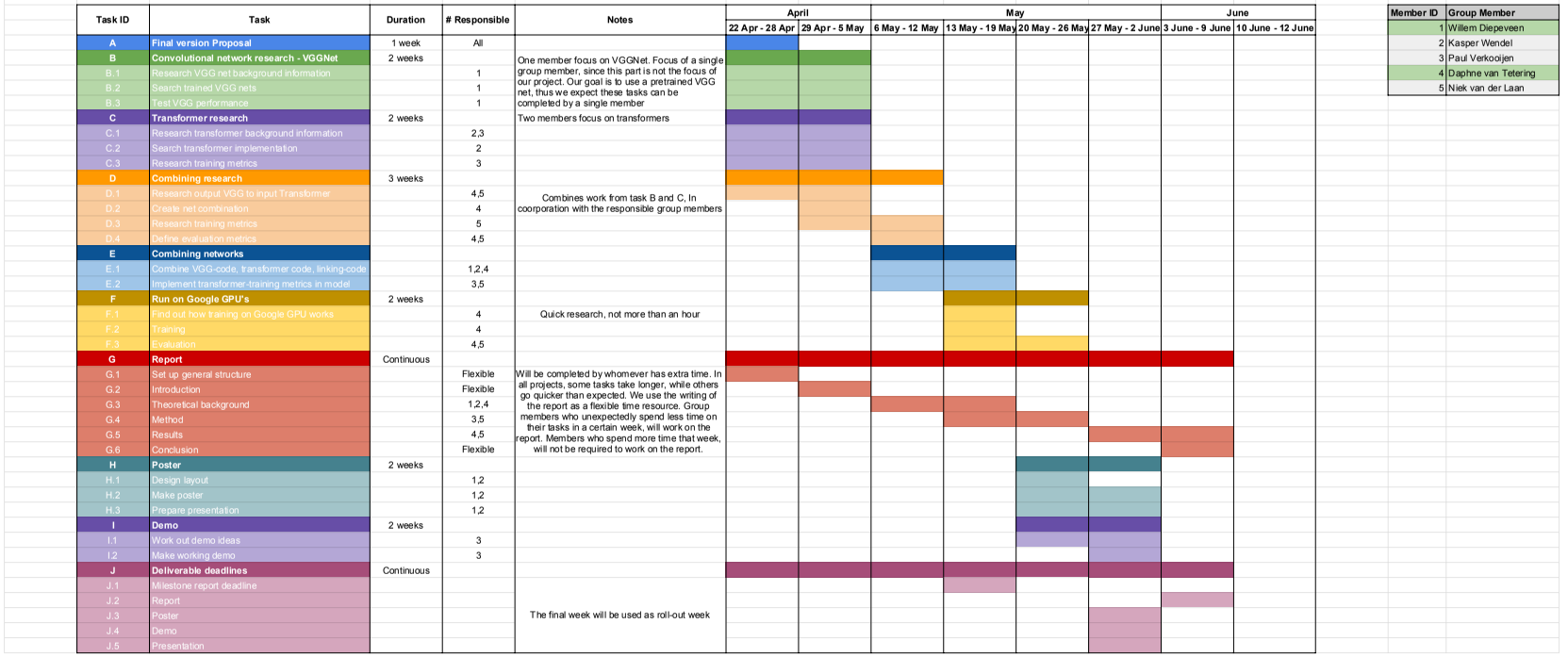
Our group has a background in Computer Science (4x) and Applied Mathematics (1x). The Computer Science students have trained a forward network once or twice. However, in the case of a large complex deep neural network architecture (which is time consuming to train), we do not have experience.

To get insight into the feasibility of our project, we consider some problems one can run into that we gave some thought. The main problems we considered were: data set, programming, complexity of model vs. running time. As described already we will be using existing data, so we know what to expect from the data and do not expect wasting time on generating data. Furthermore, the code of the networks is already available. Our task is mainly concerned with combining the code bases and making it work. Wasting time on debugging should be minimal as well. A remaining issue is the running time. We do have Google credits, but we don't know quite good how long we can run with these and how long we will need. One of our reasons to go with this problem on the transformer, is the fast training possibilities of the Transformer network.

7 Planning

In Figure 2 our planning is shown. We have 5 members in our time which will all put in approximately 8 hours per week. Right now our planning is per week.

Figure 2: Gantt Chart planning



Task ID	Task	Duration	# Responsible	Notes
A	**Final version Proposal**	1 week	All	
B	**Convolutional network research - VGGNet**	2 weeks		One member focus on VGGNet. Focus of a single group member, since this part is not the focus of our project. Our goal is to use a pretrained VGG net, thus we expect these tasks can be completed by a single member
B.1	Research VGG net background information		1	
B.2	Search trained VGG nets		1	
B.3	Test VGG performance		1	
C	**Transformer research**	2 weeks		Two members focus on transformers
C.1	Research transformer background information		2,3	
C.2	Search transformer implementation		2	
C.3	Research training metrics		3	
D	**Combining research**	3 weeks		
D.1	Research output VGG to input Transformer		4,5	Combines work from task B and C, In coorporation with the responsible group members
D.2	Create net combination		4	
D.3	Research training metrics		5	
D.4	Define evaluation metrics		4,5	
E	**Combining networks**			
E.1	Combine VGG-code, transformer code, linking-code		1,2,4	
E.2	Implement transformer-training metrics in model		3,5	
F	**Run on Google GPU's**	2 weeks		
F.1	Find out how training on Google GPU works		4	Quick research, not more than an hour
F.2	Training		4	
F.3	Evaluation		4,5	
G	**Report**	Continuous		
G.1	Set up general structure		Flexible	Will be completed by whomever has extra time. In all projects, some tasks take longer, while others go quicker than expected. We use the writing of the report as a flexible time resource. Group members who unexpectedly spend less time on their tasks in a certain week, will work on the report. Members who spend more time that week, will not be required to work on the report.
G.2	Introduction		Flexible	
G.3	Theoretical background		1,2,4	
G.4	Method		3,5	
G.5	Results		4,5	
G.6	Conclusion		Flexible	
H	**Poster**	2 weeks		
H.1	Design layout		1,2	
H.2	Make poster		1,2	
H.3	Prepare presentation		1,2	
I	**Demo**	2 weeks		
I.1	Work out demo ideas		3	
I.2	Make working demo		3	
J	**Deliverable deadlines**	Continuous		
J.1	Milestone report deadline			
J.2	Report			
J.3	Poster			The final week will be used as roll-out week
J.4	Demo			
J.5	Presentation			

Member ID	Group Member
1	Willem Diepeveen
2	Kasper Wendel
3	Paul Verkooijen
4	Daphne van Tetering
5	Niek van der Laan

References

[1] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[2] Jason Brownlee. How to Prepare a Photo Caption Dataset for Training a Deep Learning Model. `https://machinelearningmastery.com/prepare-photo-caption-dataset-training-deep-learning-model/`, November 2017. Accessed 02-04-2019.

[3] Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task: data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47(1):853–899, May 2013.

[4] Md Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, Hamid Laga, et al. A comprehensive study of deep learning for image captioning. *arXiv preprint arXiv:1810.04020*, 2018.

[5] Hsankesara. Flickr Image Dataset. `https://www.kaggle.com/hsankesara/flickr-image-dataset`, June 2018. Accessed 02-04-2019.

[6] Junqi Jin, Kun Fu, Runpeng Cui, Fei Sha, and Changshui Zhang. Aligning where to see and what to tell: image caption with region-based attention and scene factorization. *arXiv preprint arXiv:1506.06272*, 2015.

[7] Andrej Karpathy. Deep Visual-Semantic Alignments for Generating Image Descriptions. `https://cs.stanford.edu/people/karpathy/deepimagesent/`, 2015. Accessed 02-04-2019.

[8] Computer Science Department University of Illinois. Flickr 8k dataset. `https://forms.illinois.edu/sec/1713398`. Accessed 02-04-2019.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[10] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015.

[11] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057, 2015.