**Flexibility is all you need**

Brian Rikshpun

Guy Cohen

Ester Vaknin

Almog Klein

Afeka College of Engineering

January 4, 2021

**1    Abstract**

Inspired by the attention mechanism as implemented in the Transformer [1] and changed the state-of-the-art models in the famous NLP field – machine translation, our team looked up to see how attention mechanism can be integrated in other ML tasks. we choose the image classification field and we came across a model combining recent advances in the deep neural network – the option to “go deeper” by using residual neural networks [2] and their improvements – Inception [3] and ResNext [4] combined with the attention mechanism [5].

Although the architecture combining two methods that we expect will give us better scores together, we can see that it scores poorly on the CIFAR100 (79% accuracy) while preforming very good on the CIFAR10 (95% accuracy).

In this work we are investigating the flexibility of the suggested model, and how “simple” changes will affect the score of the model.

**2    Introduction**

The architecture we are examining published in 2017,two innovations in the field of deep learning are published in the past 2 years (between 2015 – 2017).

The first one is the concept of “Attention”, as shown in the article “Attention is all you need” [1], the usage of multihead attention in the architecture of the transformer is improving the score on machine translation tasks. by that we are giving the network the ability to decide on which words to focus in each time lap in the training process to obtain better scores.

The second one is the concept of the residual network, allowing us to build much deeper networks using “skip-connections” – Residual blocks [2] and “more efficient” ways of those blocks - the “ResNext” [3] and the “Inception” [4].

Combining those two strong innovations into one architecture is suggested in the “Residual attention neural network” - RANN [5] which we are examining in this work.

In the RANN, in every attention module there are two main branches: trunk branch and softmax branch. The trunk branch is a long residual connection which takes in consideration the input to the attention module. While the softmax branch is minimizing the input by pooling, to discover which features of the input we should focus on. After we get this information, we amplify the input back to its first size with interpolation, while considering the features it had in each pool corresponding with the interpolation size. At last we combine the trunk branch and the softmax branch and moving forward to the next attention module to find the next mask.

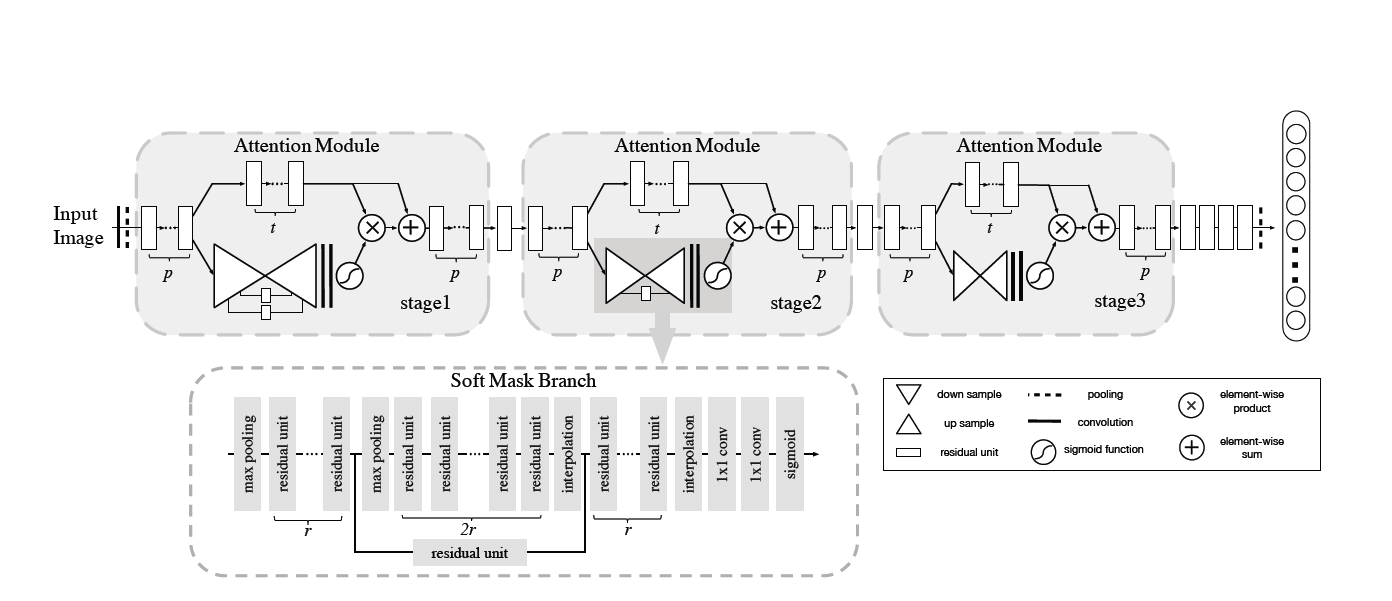


Figure 1 - Residual attention network (RANN)

**3    Data description**

In our work we used the [CIFAR10](https://www.cs.toronto.edu/~kriz/cifar.html) and CIFAR100 image datasets, the CIFAR10 is a subset of the CIFAR100 and have fewer classification options. While the first one has 10 classes and 6000 images per class, the second has 100 classes and 600 images per class (83% train).

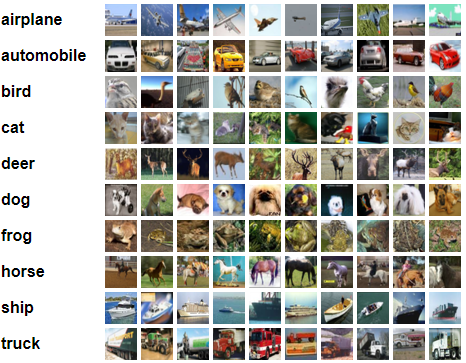


image 1 - CIFAR10 example

**4    Methodology of work**

Our work is based on a code we found online – [git](https://github.com/tengshaofeng/ResidualAttentionNetwork-pytorch).

First we had to learn the PyTorch library to get better with the massive code we found.

Our goal in this work is to find out how much the suggested architecture (RANN) is “flexible”, how changes will affect the score on CIFAR10 and CIFAR100.

The changes we made (innovations) are:

1. The code got from GitHub was tested on CIFAR10, we tested the model on CIFAR100.
2. Instead of 3 attention modules we tested the model with 1,2 and 4 attention modules on the CIFAR100 and 2 and 3 attention modules on the CIFAR10.

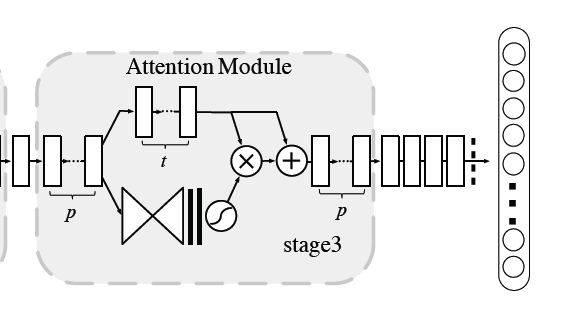
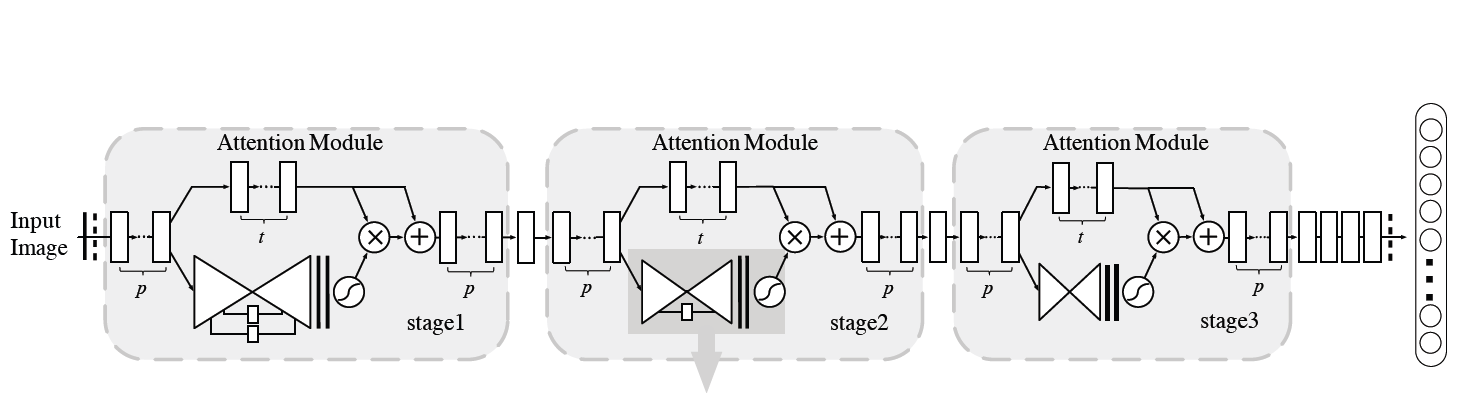
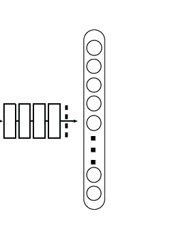
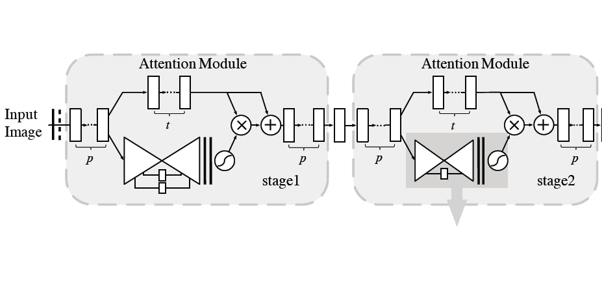
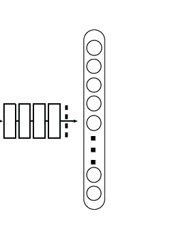
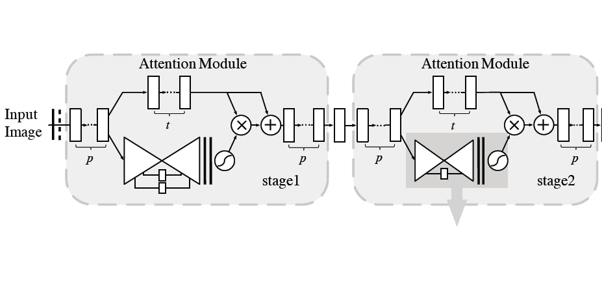


Figure - RANN with 1,2 and 4 attention modules

As suggested in the article, we saved the parameters p, t and r to be 1,2 and 1 respectively,

Each training session had 300 epochs.

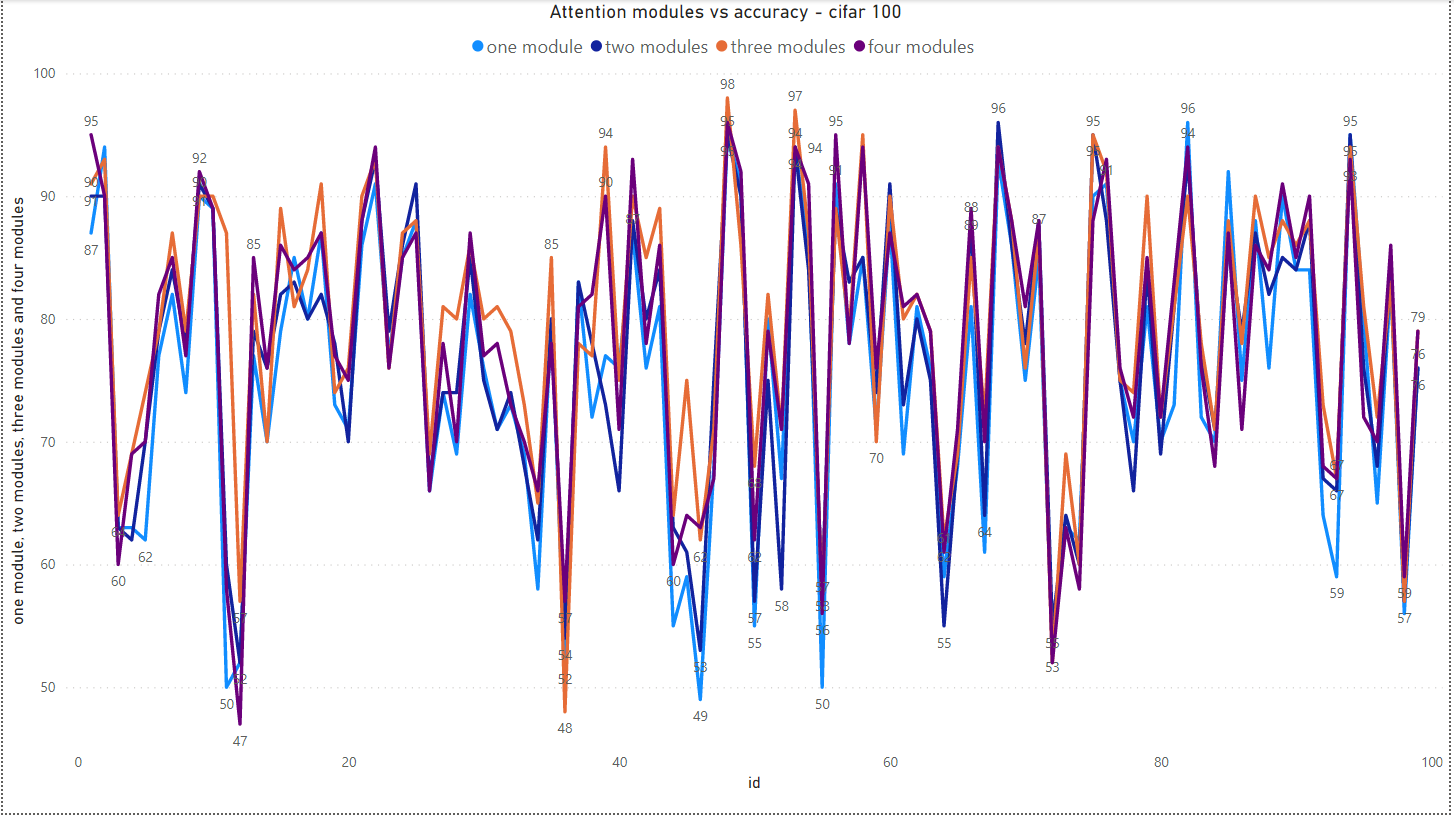
**5    Results**

Figure 3 - Accuracy of each class in the CIFAR100 with different number of attention modules (Azure – 1, Blue – 2, Orange – 3, Purple - 4)

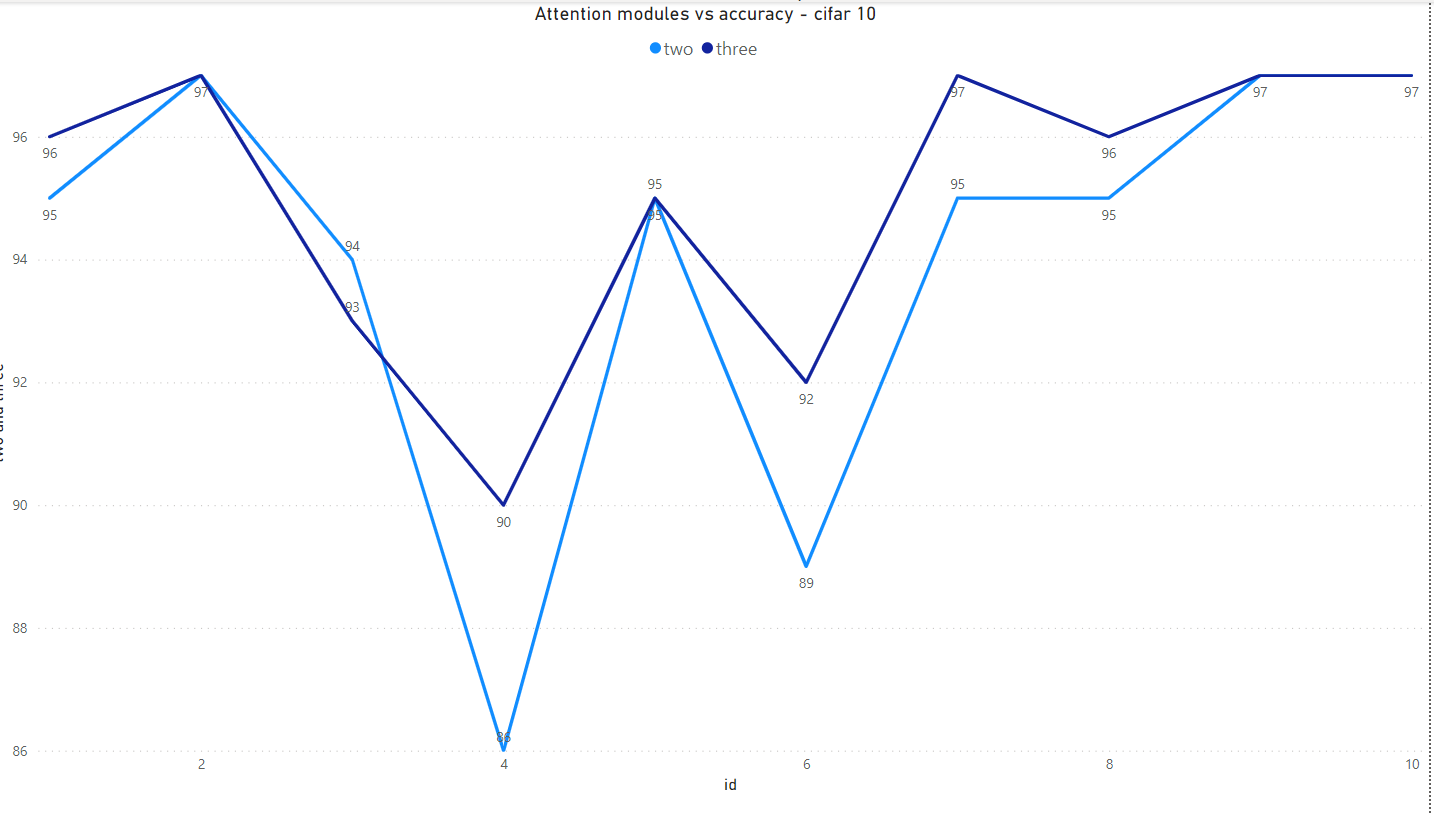


Figure 4 - Accuracy of each class in the CIFAR10 with different number of attention modules (Azure – 2, Blue – 3)

|  |  |  |
| --- | --- | --- |
|  | Number of attention modules | Accuracy |
| CIFAR100 | 1 | 75.34% |
| 2 | 77.10% |
| 3 | 79.28% |
| 4 | 78.34% |
| CIFAR10 | 2 | 94.31% |
| 3 | 95.40% |

Table 1 - Accuracy results

**6    Conclusions**

In this work we investigated the flexibility of the suggested architecture (RANN) and how changes will affect the score of the model on CIFAR10 and CIFAR100.

The results are slightly against the expectation we had, we thought that if we will add one more attention module the accuracy will raise but the opposite happened. We also thought that if we will subtract attention modules the accuracy will drop, which did happen but we expected to see much larger drop and not a decrease in few percentages.

We can see in clear way that there is no significant difference between the accuracy using different number of attention modules in the architecture (on both CIFAR10 and CIFAR100). Using one attention module yields us 75% accuracy while four attention modules yields us 78% (on the CIFAR100).  
We can assume that using only one attention module is enough because the dataset we used to train and test the model contains images with relatively low number of features, so we can locate the needed features to the right classification using only one attention model in most of the cases and there is no extra need in more attention modules to find other masks.

After analyzing the results, we can’t tell clearly that the architecture is “flexible” on a noisy datasets containing more challenging pictures than the CIFAR’s, but we can tell that on the CIFAR data sets the architecture is flexible which allows us to build model with fewer attention modules without hurting too much to the accuracy and of course winning the training time reduction in this trade-off.

**7   Reference**

[1] Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems*. 2017. <http://papers.nips.cc/paper/7181-attention-is-all-you-%0Aneed.pdf>

[2] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. <http://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html>

[3] Szegedy, Christian, et al. "Going deeper with convolutions." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015. <https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html>

[4] Xie, Saining, et al. "Aggregated residual transformations for deep neural networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017. <http://openaccess.thecvf.com/content_cvpr_2017/html/Xie_Aggregated_Residual_Transformations_CVPR_2017_paper.html>

[5] Wang, Fei, et al. "Residual attention network for image classification." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017. <http://openaccess.thecvf.com/content_cvpr_2017/html/Wang_Residual_Attention_Network_CVPR_2017_paper.html>