ANALYZING MEMORIZATION IN VISUAL AUTOREGRESSIVE MODELS

Johannes Reber
johannes.reber@pm.me

ABSTRACT

Visual autoregressive models are generative neural networks for creating images. We analyze the memorization and find differing results for the autoencoder and the transformer model. The encoder represents more than 90% of the dataset in the last five layers, whereas very few datapoints have a strong representation in the beginning of the encoder. The transformer stores most information in the middle of its architecture. Generally, the classes are barley distributed in the layers and memorization can be found after consecutive activation layers.

1 MEASURING MEMORIZATION IS INSIGHTFUL

Generative Neural Networks have become a big center of attention in recent years. Chatbots like ChatGPT and their respective image generators like DALL-E produce results that are nearly indistinguishable from human work. The biggest challenges in further improving these models are scalability and their zero-shot generalization ability. However, more neurons do not always achieve better results. It is therefore important to explore different models and strategies to improve results. Visual Autoregressive Models (VAR) proposed by Tian et al. (2024) are a novel approach for generating images. We will explore the memorization of the VAR models by measuring the activation of each layer. This provides us with insights into the workings of the network. We can explore pruning options to further optimize the model so that it can be more efficient. Furthermore, we can see how well the network remembers single datapoints. From this we could assess the feasibility of privacy leakages like a model inversion attack.

2 RELATED WORK

VAR models are a novel image-generating approach proposed by Tian et al. (2024). In contrast to token-based predictors like GPT, it can be described as a next resolution predictor divided into two stages. The first stage includes an autoencoder that encodes the training images into a token map with which the second stage of training is conducted. The second stage contains a transformer model that predicts the next token in the series. This leads to an upscaled version of an image that provides high zero-shot generalization and high scaling laws. This makes the model a faster and better performing alternative to other image generators.

Measuring memorization was proposed by Wang et al. (2024). During a forward pass, memorization can be measured per unit, i.e., per neuron or convolution. The memorization observes how much a unit "fires" by observing the activation function. First, the the mean activation over multiple augmentations of a datapoint x on unit u is determined as

$$\mu_u(x) = \mathbb{E}_{x' \sim \text{Aug}(x)} \text{ activation}_u(x'). \tag{1}$$

Then the maximum mean activation $\mu_{max,u}$ and the mean activity $\mu_{-max,u}$ are calculated. The unit memorization score is calculated as

$$\text{UnitMem}(u) = \frac{\mu_{max,u} - \mu_{-max,u}}{\mu_{max,u} + \mu_{-max,u}}. \tag{2}$$

It was found that single data points are memorized throughout all layers of an encoder, especially in single units. Additionally, memorization is mainly found in the fully connected layers of the transformer models.

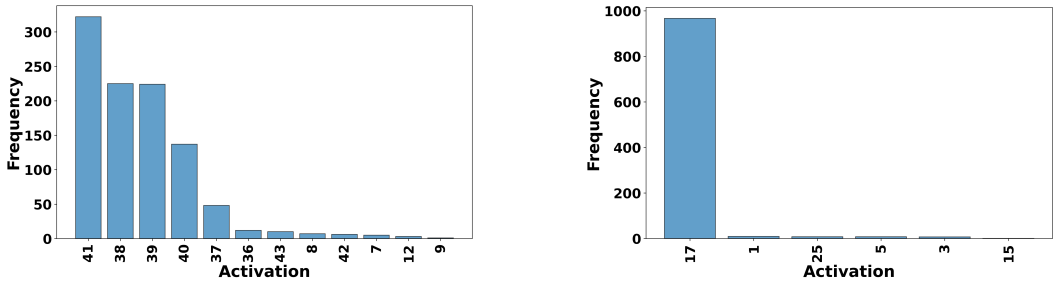

Figure 1: Most memorizing units by layer in the encoding model (left) and transformer models (right).

3 Measuring Memorization on VARd16 models

To measure the memorization, we closely follow the work of Wang et al. (2024). We start by selecting a subset of the training data on which we want to perform the memorization evaluation. We calculate the mean activation as described in Equation 1 after applying the augmentations on the images. We continue to calculate the unit memorization scores for each data point in each layer according to Equation 2.

4 Evaluation

The experiments were carried out on a GeForce GTX 1060 with 6GB VRAM using the shallowest VAR model with 16 self attention layers with in total 33 activation functions. The respective autoencoder has a total of 87 activations. The decoder is neglected for this work since it does not have an influence. We selected a subset of ImageNet, with each class represented once for manageable performance. As augmentations, we applied random grayscale conversion, flipping, affine translation by 10%, and color jitter [1]. We evaluate the data by two criteria. For the first we find the most active unit for each class. The results for the encoding model and transformer model are shown in Figure 1. We notice that most classes are activated at the end of the encoder or in the middle of the transformer. Furthermore, this does not allow us to find pattern in this distribution. We additionally analyze the variance of activation in each layer. Some classes show high activities in these layers, whereas other do not lead to high variances. For the maximum measured activation we note that there is no similar class in the dataset. We can conclude that indeed some datapoints are induce stronger signals in the networks than others.

5 Conclusion & Limitations & Future Work

Our findings indicate that the most memorizing layers in a VARd16 model are located at the end of the encoder and in the middle of the transformer, with some outliers in the encoder. However, the variances between activations are small indicating that some datapoints produce stronger signals than others. These results generally align with the observations made by Wang et al. (2024) on different self-supervised learning models.

However, our approach has certain limitations. It requires prior knowledge of the training dataset and the model architecture. Additionally, depending on the size of the data set, the method may require substantial data. Furthermore, since training is not fully deterministic, results may vary between different runs.

For future work, deeper models could be analyzed to gain further insight into memorization patterns. In addition, larger datasets could be explored to improve generalization of the approach. Finally, applying the LayerMem metric from Wang et al. (2024) could provide a more detailed evaluation of memorization behavior across layers. To check whether classes that reach a high memorization score in the same layer show similar patterns, a clustering could be applied.

[1] The code to produce the results can be found at `https://github.com/j-reber/VAR-analysis`

REFERENCES

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual Autoregressive Modeling: Scalable Image Generation via Next-Scale Prediction. June 2024. URL http://arxiv.org/abs/2404.02905.

Wenhao Wang, Adam Dziedzic, Michael Backes, and Franziska Boenisch. Localizing Memorization in SSL Vision Encoders. November 2024. URL https://openreview.net/forum?id=R46HGlIjcG.