Numerical Evidence That the Power of Artificial Neural Networks Limits Strong AI

Roman Englert roman.englert@fom-net.de
FOM University of Applied Sciences,
Computer Science, Herkulesstraße 32, 45127 Essen, Germany.
Siegen University, Faculty III, New Media and Information Systems,
Kohlbettstraße 15, 57072 Siegen, Germany.
Jörg Muschiol jörg.muschiol@fom-net.de
FOM University of Applied Sciences,
Computer Science, Herkulesstraße 32, 45127 Essen, Germany.

Corresponding Author: Roman Englert.

Abstract

A famous definition of AI is based on the terms weak and strong AI from McCarthy. An open question is the characterization of these terms, i.e., the transition from weak to strong. Nearly no research results are known for this complex and important question. In this paper we investigate how the size and structure of a Neural Network (NN) limits the learnability of a training sample, and thus, can be used to discriminate weak and strong AI (domains). Furthermore, the size of the training sample is a primary parameter for the training effort estimation with the big O function. The needed training repetitions may also limit the learning tractability and will be investigated. The results are illustrated with an analysis of a feedforward NN and a training sample for language with 1,000 words including the effort for the training repetitions.

Keywords: Power of AI, Weak and strong AI, NN, Betti numbers, Training Repetitions of NN, Training Sample, Dimension of NN.

1. INTRODUCTION

Artificial Neural Networks (NN) have a limitation in their power, since the number of neurons that enable the storing of information is bounded. The challenge is twofold: First, to describe the computational limitation of a NN, and second, what is the maximum complexity of a domain that can be learned by a NN? Instead, the human brain and its areas are considered in order to analyze how much effort must be spent to train a certain brain area. The human brain has dedicated areas for thinking, sensing and perceiving like logic thinking, speech, or for planning [1-3]. Liu states that weak AI ”represent computational systems that exhibits as if they own human intelligence, but they do not” [4]. The aforementioned brain areas are partitions of the brain and their neurons are effectively connected with each other neurons, i.e., each neuron is in average with 1,000 other

Citation: Roman Englert and Jörg Muschiol. Numerical Evidence That the Power of Artificial Neural Networks Limits Strong AI. Advances in Artificial Intelligence and Machine Learning 2022;2(2):22.

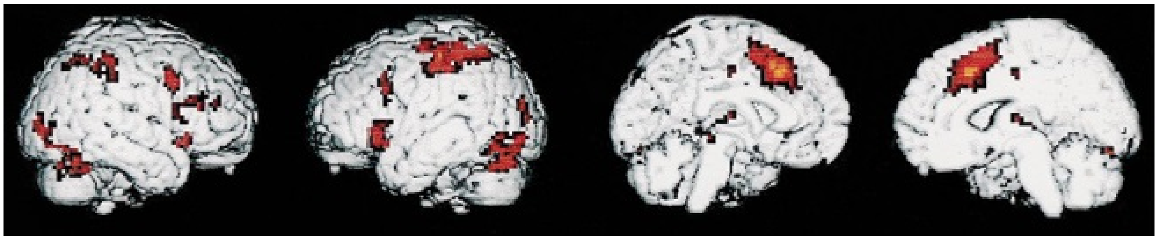

Figure 1: Activated brain areas for human language processing comparing words with letter stings 'X' [5].

neurons connected. As an example consider FIGURE 1, showing the activated brain areas for human language processing comparing words with letter stings 'X': left and right inferior and frontal lobe, the left precentral gyrus, left and right upper and lower parietal regions, the left and right occipital regions, and the SMA and upper cerebellar regions [5].

The approach is to interpret a brain area as a domain, resp. model, for training a NN. And the goal is to determine the computational effort for learning this model. This complexity can be estimated using Betti numbers: Betti numbers are used in algebraic topology to analyze the connectivity of simplicial complexes like lines and holes[6]. They represent invariants about the surface, resp. structure of a NN. Knowing the complexity for learning a model, the number of required neurons to learn a model can be estimated and applied to the complexity in order to determine whether the domain is computationally tractable or not [7], which may limit the power of weak AI [8,9], where machines are assumed to be able to think and have genuine understandings. Furthermore, domains, i.e. training data for NN, can grow by $O(n^2)$ due to bias mitigation, where n denotes the number of features [10]. As brain area the human speech is considered assuming that we use 1,000 words with in average six different meanings. For this domain the computational effort to train a feedforward NN is analyzed and discussed.

The paper is organized as follows: Section 2 contains upper bounds for the number of hidden layers and hidden neurons of a NN. Subsequently, the complexity of NNs based on Betti numbers is described in Section 3. The number of required training repetitions is highlighted in Section 4. Then, in Section 5 an example for the required size and the training effort of a feedforward NN for language is calculated simulating the brain access for language understanding. The conclusion contains an outlook with description of future research steps.

2. MODELING A NN BASED ON UPPER BOUNDS FOR THE NUMBER OF HIDDEN NODES

Maier and Dandy (2001) investigate the number of hidden layers and upper bounds for the number of hidden neurons of a NN. FIGURE 2 shows for illustration a NN with one hidden layer and FIGURE 3 a NN with more than one hidden layer, here three hidden layers. They describe that any continuous (learning) function for a multi-layer NN can be approximated by a NN with one hidden layer (given sufficient degrees of freedom) [11]. Additionally, they state that the use of two

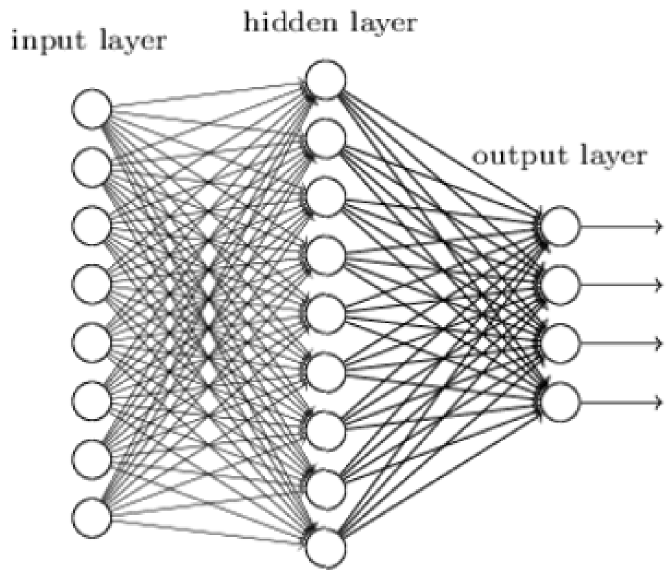

Figure 2: Neural Network with one hidden layer.

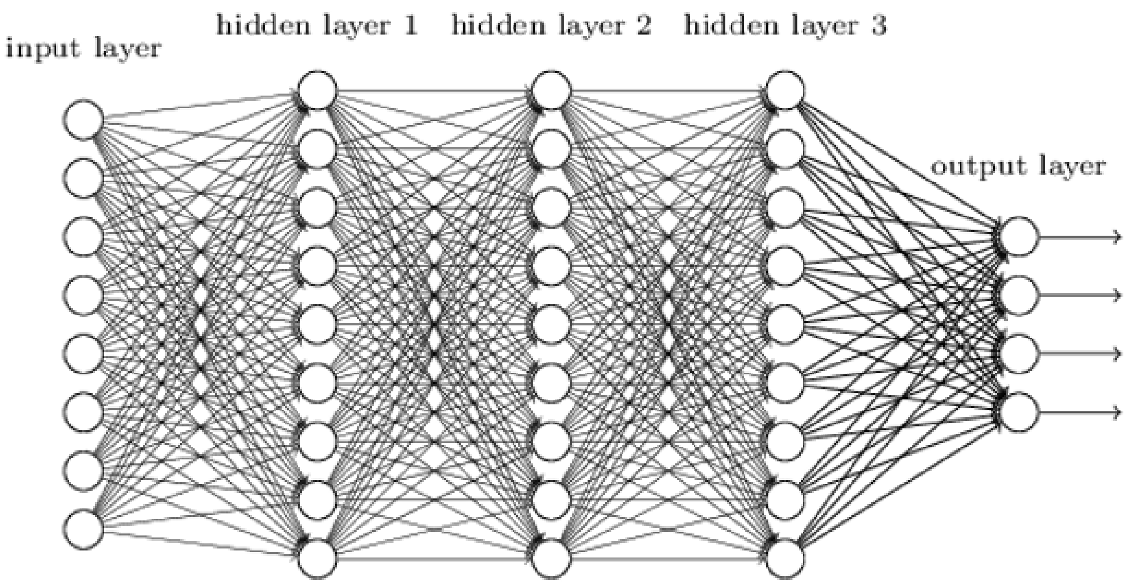

Figure 3: Neural Network with more than one hidden layer, here three hidden layers.

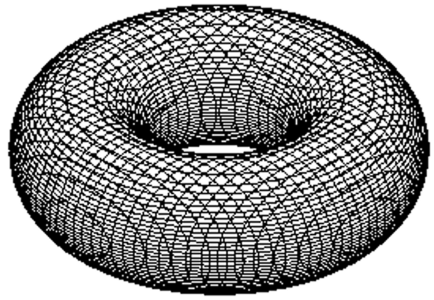

Figure 4: Three-dimensional torus to illustrate Betti numbers.

hidden layers is only justified for ”the most esoteric applications”. Further reading can be done in the work of Maier and Dandy [p. 676][11]. They provide two upper bounds for the number of hidden neurons:

$$n_h \leq 2 \cdot n_i + 1 \tag{1}$$

and

$$n_h \leq \frac{n_{tr}}{n_i + 1} \tag{2}$$

with n_h is the number of hidden neurons, n_i is the number of input neurons, and n_{tr} is the number of training samples. For performance reasons the smaller number obtained from (1) and (2) should be taken, or even fewer. Note, these bounds help to prevent over-fitting, since oversized NNs tend to learn a domain by heart, and as a consequence, they are weak predictors.

3. ANALYSIS OF THE COMPLEXITY FOR TRAINING A NN USING BETTI NUMBERS

From the former section an upper bound for the number of hidden neurons is known depending on the number of input neurons (Equation 1) and for Equation 2 the size of the training sample. The core question to be answered is twofold, namely which number of hidden neurons is (a) required for training a NN given an domain (training sample) and (b) computationally tractable for a NN? For the analysis feedforward NNs are considered. A well-investigated measure for the complexity of a NN are Betti numbers [6]. These numbers are known from algebraic topology and are applied to characterize NNs. Betti numbers are used to describe spaces with different topological properties. More formally, assume a feedforward NN $\mathcal{N}$ that implements a function $f_{\mathcal{N}} : \mathcal{R}^2 \rightarrow \mathcal{R}$, then the complexity of the function $f_{\mathcal{N}}$ is given by the topological complexity of the set $S_{\mathcal{N}} = \{x \in \mathcal{R}^n | f_{\mathcal{N}}(x) \geq 0\}$, with $S_{\mathcal{N}}$ contains only positively classified patterns [6]. Then, for any subset $S_{\mathcal{N}} \subset \mathcal{R}^n$ exist n Betti numbers

$$b_i(S_{\mathcal{N}}), 0 \leq i \leq n - 1. \tag{3}$$

The first Betti number $b_0(S_{\mathcal{N}})$ can be interpreted as the number of connected components of $S_{\mathcal{N}}$, and the i-th Betti number $b_i(S_{\mathcal{N}})$ represents the number of $(i + 1)$-dimensional holes in $S_{\mathcal{N}}$. For example a three-dimensional torus T (FIGURE 4) has the Betti numbers $b_0(T) = 1$ (number of

Figure 5: Letters ’A’ to ’F’ with three dimensions for the illustration of Betti numbers.

Table 1: Upper bounds, depending on the activation function of a NN, the number n_i of input neurons, and the number n_h of hidden neurons [6].

Activation function	Bound
Threshold	$O({n_h}^{n_i})$
Arctan	$O((n_i + n_h)^{n_i+2})$

connected components), $b_1(T) = 2$ (number of two-dimensional or ”circular” holes), $b_2(T) = 1$ (number of three-dimensional ”voids” or ”cavities”), and for $i > 2$ is $b_i(T) = 0$.

For compact manifolds like the torus is the sequence of Betti numbers zero from some point onward, since Betti numbers vanish above the dimension of a space, and thus, they are all finite (Equation 3).

As second example, shows FIGURE 5, the three-dimensional letters ’A’ to ’F’: the Betti numbers for ’A’ and ’D’ are the same as for the torus, since they also have one hole. For the letter ’B’ holds also $b_0(T) = 1$ (one connected component), $b_1(T) = 3$ (number of circular holes with both holes separately and the two holes together), and $b_2(T) = 2$ (three-dimensional cavities). And the letters ’C’, ’E’ and ’F’ have the Betti numbers $b_0(T) = 1$, and no holes, so $b_1(T) = b_2(T) = 0$. For all letters hold $b_i(T) = 0$ with $i > 2$. For the string ’AB’ holds $b_0(T) = 2$, since the letters ’A’ and ’B’ of the string are disconnected.

The total complexity of the space $S_\mathcal{N}$ is the sum of the Betti numbers $B(S_\mathcal{N}) = \sum_i b_i(S_\mathcal{N})$. Then, upper bounds for the structure of $\mathcal{N}$ can be proven (TABLE 1) [6], depending on the activation function of the NN $\mathcal{N}$, the number n_i of input neurons, and the number n_h of hidden neurons. For the number l of hidden layers is $l = 1$ assumed, as argued in the former section.

TABLE 1 indicates that for feedforward NNs with threshold-based and arctan activation functions the upper bounds for the complexity are polynomial, in case for one hidden layer (aka shallow networks). As a simple example for illustration, consider the NN in FIGURE 2, with $n_i = 8$ input neurons and $n_h = 9$ hidden neurons: with a threshold activation function the Betti number complexity of the NN structure is bound by the constant $O(9^8)$ for threshold-based activation functions. Maier and Dandy (2001) showed, that for several hidden layers ($l > 1$), so-called deep networks, the complexity can grow exponentially. However, the polynomial for shallow NN depends on the number of input neurons n_i. The tractability of polynomials with bigger n_i is exemplary considered in Section 5.

4. ESTIMATION OF THE REQUIRED TRAINING REPETITIONS FOR A NN

To avoid over-fitting upper bounds for the number of hidden neurons (Equation 1 and Equation2) are presented in Section 2. Additionally, to ensure a proper NN training, Iyer and Rhinehart (1999) developed a performance optimizer to find the best model with the lowest error, based on the lowest sum-of-squared errors of a training data set: the idea is to train the NN with random samples (with replacement) and to measure the training errors [7]. Then, order the measured errors from low to high and create a probability distribution. The probability that any single optimization has an error value x less than or equal to a is $F_X(a) = \int_0^a f_X(x)dx$. $F_X(a)$ is the cumulative distribution function (CDF) of a with $f_X(x)$ which is the probability distribution function of the error values $x \in X$. From F the CDF can be derived for the best trial from k independent trials:

$$F_k(a) = 1 - (1 - F_X(a))^k \tag{4}$$

Now, if someone decides on the confidence level ($F_k(a)$) and the percentage vicinity of the lower tail of the CDF, then Equation 4 can be rearranged to find the number k of required training repetitions:

$$k = \frac{ln(1 - F_k(a))}{ln(1 - F_X(a))} \tag{5}$$

Note, Equation 5 is independent of the domain (data set), the NN size and structure, the activation function, etc. These values are implicit contained in $f_X(x)$ resp. $F_X(a)$. Additionally, if the lower x% of all possible errors (i.e., x random training repetitions) is not close to the global optimum, then Equation 5 can be used to determine the number of random training repetitions to achieve a trained NN with a particular percentile of generalization errors. As an example consider $F_X(a) = 0.02$ (the best 2% values of all errors for the sum-of-squared-errors) with $F_k(a) = 0.99$ (99% confidence), then

$$k = \frac{ln(1 - 0.99)}{ln(1 - 0.02)} \cong 228 \tag{6}$$

random training repetitions are required. This formula (Equation 6) serves for the investigation example in the following section.

5. INVESTIGATION OF A FEEDFORWARD NN FOR MODELING THE HUMAN LANGUAGE PROCESSING

The human brain contains an area for language and reading, and some smaller used (activated) parts (FIGURE 1) [5,12,13]. In the following application of the former results, the dimension of an artificial NN is calculated to simulate the human brain area for language.

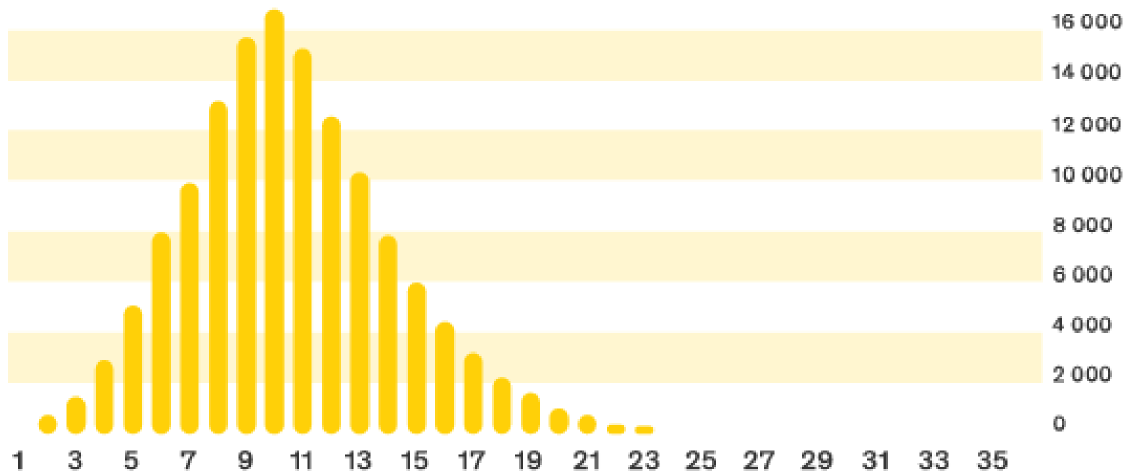

Figure 6: Average word length of the German language [14].

For the investigation we assume that humans use in average 1,000 words for communication, each word with up to six different meanings, resp. contexts, resulting in 6,000 words. The positive training data consist of these 6,000 positive examples, and additionally of the same number of negative examples with wrong meanings, these may also include antonyms. Thus, the sample data n_{tr} contain in total 12,000 training patterns (positive and negative).

FIGURE 6 shows the average word length with 10.6 characters of the German language, with the x-axis denoting the number of characters per word and the y-axis denoting the number of the most frequently used words in the German language [14]. If 1,000 words are randomly picked, then all words with a length up to 23 are candidates for a training pattern. Thus, $n_i = 23 + 1 = 24$ input neurons are needed ('+1' to distinguish positive and negative patterns). Then, Equation 1 provides the upper bound for the number of hidden neurons with $n_h \leq 2 \cdot 24 + 1 = 49$. Maier and Dandy (2001) provide also an upper bound dependent on the sample size (Equation 2) to avoid over-fitting: $n_h \leq \frac{12{,}000}{24+1} = 480$. Here, n_h is large due to the size of the training sample.

The Betti numbers for the complexity of the NN (with activation function arctan) are bounded by $O((n_i + n_h)^{n_i+2}) = 73^{26} = 7,9644e+48$ (FIGURE 7). This huge number justifies scepticism for the computational tractability of the NN and the training sample. The required training repetitions of 228 seem to be moderate for the 2% best values (Equation 6). Language understanding and speaking are only some parts of the human brain that contains also many other areas like perception, thinking and sensing.

6. CONCLUSION AND OUTLOOK

This paper considers and applies upper bounds for the number of hidden neurons in a single-layer NN. The Betti numbers provide a measure for the effort to train a NN, e.g., polynomial in big omicron for a feedforward NN with the activation function arctan. They are invariants about the surface, resp. structure of a NN. Even this result is polynomial, the degree of the polynomial depends

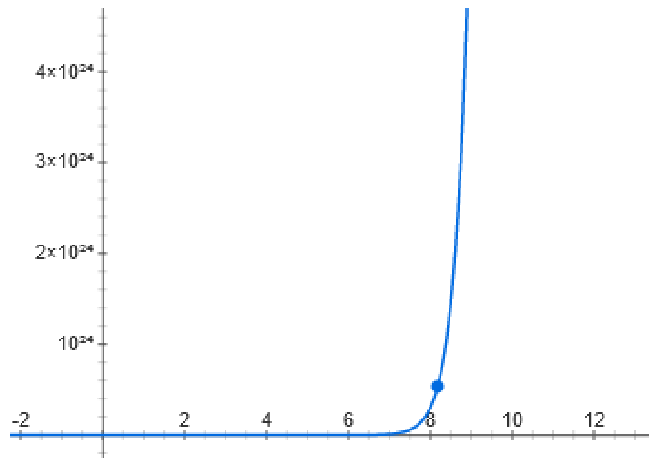

Figure 7: Plot of the polynomial $f(x) = x^{26}$ with degree 26.

strongly on the number of hidden neurons, and thus, the computational tractability for larger NN is questionable. This result limits already weak AI numerically, which has been applied (only) to parts of the human brain. To train a NN the expected error is aimed to be minimized, and thus, this requires training repetitions which may have to be repeated several hundreds times. The expected number of repetitions can be calculated based on the cumulated error distribution.

Some questions are still open, despite for the complexity analysis with Betti numbers: Which dimensions of a NN are in practice computational tractable? Domains, i.e. training data for NN, can grow by $O(n^2)$ due to bias mitigation, where n denotes the number of features [10]. Another aspect is: do the theoretical bounds allow a complete modeling and simulation of parts of the human brain without severe capability restrictions? Especially, the latter question requires a deeper understanding and investigation of the human brain functionality, even with a simplification in order to enable an artificial modeling. As a next step the enhancement of domains (training data) by bias mitigation knowledge to achieve stronger AI will be investigated.

7. CONFLICT OF INTEREST

The authors declare no competing interests.

8. ACKNOWLEDGEMENT

Many thanks to Alexander Röser for fruitful discussions.

References

[1] Grasby P, Frith C, Friston K, Bench C, Frackowiak R, et al. Functional Mapping Of Brain Areas Implicated in Auditory—Verbal Memory Function. Brain.1993;116:1–20.

[2] Raichle ME, Snyder AZ. A Default Mode of Brain Function: A Brief History of an Evolving Idea. Neuroimage.2007;37:1083–1090.

[3] Nieuwenhuis S, Heslenfeld DJ, Von Geusau NJA, Mars RB, Holroyd CB, et al. Activity in Human Reward-Sensitive Brain Areas Is Strongly Context Dependent. Neuroimage.2005;25:1302–1309.

[4] https://arxiv.org/abs/2103.15294

[5] Jessen F, Erb M, Klose U, Lotze M, Grodd W, et al. Activation of Human Language Processing Brain Regions After the Presentation of Random Letter Strings Demonstrated With Event-Related Functional Magnetic Resonance Imaging. Neuroscience Letters.1999;270:13–16.

[6] Bianchini M, Scarselli F. On the Complexity of Neural Network Classifiers: A Comparison Between Shallow and Deep Architectures. IEEE Transactions on Neural Networks and Learning Systems.2014;25:1553–1565.

[7] Iyer MS, Rhinehart RR. A Method to Determine the Required Number of Neural-Network Training Repetitions. IEEE Transactions on Neural Networks.1999;10:427–432.

[8] Norvig PR, Intelligence SA. A Modern Approach. Prentice Hall. Upper Saddle River, NJ, USA. 2002.

[9] Al-Rifaie MM, Bishop M. Weak and Strong Computational Creativity. In: Computational Creativity Research: Towards Creative Machines. Springer. 2015:37–49.

[10] Englert R., Muschiol J. Training Data Improvement by Automatic Generation of Semantic Networks for Bias Mitigation. American Journal of Information Science and Technology. 2022; 6:1-7.

[11] Mayer HR, Dandy GC. Neural Network-Based Modelling of Environmental Variables: A Systematic Approach. Mathematical and Computer Modelling. 2001; 33:669–682.

[12] Leiner HC, Leiner AL, Dow RS. Cognitive and Language Functions of the Human Cerebellum. Trends in Neurosciences. 1993;16, 444- 447.

[13] Raichle ME, Snyder AZ. A Default Mode of Brain Function: A Brief History of an Evolving Idea. Neuroimage. 2007; 37:1083–1090.

[14] https://www.duden.de/sprachwissen/sprachratgeber/Durchschnittliche-Lange-eines-deutschen-Wortes