

ANALYSIS OF MACHINE LEARNING APPLIED TO BOARD GAMES

ADAM AMANBAEV, JONATHAN HALLSTRÖM, ALVAR EDVARDSSON
HUGO ÅKERFELDT, ROMEO PATZER

SUPERVISOR: ULF BACKLUND

ABSTRACT

In October 2022 DeepMind announced an artificial intelligence in a preprint [1] that was called AlphaTensor. It utilized *artificial neural networks* in combination with *heuristic algorithms* in order to solve the unsolved problem of finding faster algorithms for matrix multiplication. AlphaTensor was succesful and has shed light on the process of finding new algorithms. In this paper we aim to study the effectiveness of different *neural networks* and *heuristic algorithms* such as the one used in AlphaTensor. More precisely, we intend to analyze the efficiency of those networks and algorithms in combination with varying *optimizations, parameters, hyperparameters* and *architectures* applied to the classic games *Connect Four* and *Othello*.

KEYWORDS — Machine Learning, Supervised Learning, Reinforcement Learning, Neural Network, Deep Learning

PREFACE

The only true wisdom is in knowing you know nothing.

–Socrates

BACKGROUND

The concept of artificial intelligence was only popularized during the first half of the 20th century through science fiction. Computers before 1950 lacked key prerequisite for intelligence: they could not store commands, only execute them immediately. Allen Newell, Cliff Shaw and Herbert Simon created in 1956 what is often considered the first artificial intelligence program and it catalyzed the next 20 years of research in this field. Gordon Moore predicted in 1965 that the memory and speed of computers would double every year and the prediction, known as Moore’s law, has been true ever since. The growth led to new possibilities for the field of artificial intelligence and many landmark goals were achieved during the 1990s and 2000s. In 1997 reigning world chess champion and grandmaster Gary Kasparov was defeated by IBM’s computer program Deep Blue [2]. 19 years later Deepmind’s artificial intelligence AlphaGo [3] beat the South Korean grandmaster Lee Sedol regarded as one of the best Go players of all time. Go is due to its large search space considerably more complex than chess and AlphaGo taught itself how to play without any proprietary knowledge in comparison to Deep Blue that was heavily influenced by the pre-coded strategies. Shortly after the success of AlphaGo Deepmind began working on protein structure prediction. The artificial intelligence AlphaFold [4] is now by a huge margin the most accurate in predicting protein structures and it is accelerating research in nearly every field of biology. As this field of research is becoming a substantial part of today’s science we find it important to analyze different aspects of it.

SUMMARY OF NOTATION

I am a forest, and a night of dark trees: but he who is not afraid of my darkness, will find banks full of roses under my cypresses.

— Friedrich Nietzsche, Thus Spoke Zarathustra

$\operatorname{argmax}_x f(x)$	$\{x \mid f(x) = \max_{x'} f(x')\}$
$\operatorname{argmin}_x f(x)$	$\{x \mid f(x) = \min_{x'} f(x')\}$
\leftarrow	assignment
$P(a \mid b)$	probability of a given b

CONTENTS

Summary of Notation	i
§1 Introduction	1
§1.1 Background	1
§1.2 Analysis	1
§1.3 Delimitations	1
§1.4 Division of Work	2
§2 Games	3
§2.1 Connect Four	3
§2.2 Othello	3
§3 Deep Learning	4
§3.1 Introduction	4
§3.2 Basic Architecture of Neural Networks	5
§3.2.1 Perceptron	5
§3.2.2 Fully Connected Neural Network	5
§3.3 Backpropagation	6
§3.3.1 Gradient Descent	6
§3.3.2 Loss Function	6
§3.4 Complications with Neural Network Training	7
§3.4.1 Overfitting	7
§3.4.2 Vanishing and Exploding Gradient	7
§3.4.3 Difficulties in Convergence	7
§3.4.4 Local Optima	7
§3.4.5 Computational Challenges	7
§3.5 Training Neural Networks	8
§3.5.1 Backpropagation in Detail	8
§3.5.2 Preventing Extreme Gradients	8
§3.5.3 Gradient Descent Strategies	8
§3.6 Generalizing Neural Networks	9
§3.6.1 Regularization	9
§3.6.2 Dropout	9
§3.7 Convolutional Neural Network	10
§3.7.1 Historical Background	10
§3.7.2 Architecture of a Convolutional Network	10
§3.7.3 Training a Convolutional Network	10
§3.7.4 Applications of Convolutional Networks	10

§4 Reinforcement Learning	II
§4.1 Introduction	II
§4.2 Monte Carlo Tree Search	II
§4.2.1 Upper Confidence Bound	II
§5 Heuristic Algorithms	12
§5.1 MiniMax	12
§5.2 AlphaBeta Pruning	12
§6 Method	13
§6.1 Implementation	13
§7 Results	14
§8 Conclusions	15
A Appendix	16
A.1 Source Code	16

§1. INTRODUCTION

A thinker sees his own actions as experiments and questions—as attempts to find out something. Success and failure are for him answers above all.

— Friedrich Nietzsche

§1.1 BACKGROUND

Artificial neural networks [5] are computing systems consisting of connected nodes called artificial neurons and are inspired by biological neural networks that constitute human brains. The artificial networks have recently been used in different forms in order to solve various complex and abstract problems. A recent example is Deepmind’s AlphaTensor [1] which in October 2022 found a faster algorithm for matrix multiplication.

The greatest factors of an artificial neural network’s success is its architecture and how it is applied to the problem presented. AlphaZero [6] and AlphaGo [3] are computer programs whose artificial neural networks only differ in the method of application. Even though AlphaZero uses the same type of network it beats AlphaGo 100 to 0 in the board game Go which demonstrates how delicate artificial networks are. In this project we take a closer look at the intricacies of artificial neural networks with the help of two board games, Connect Four and Othello.

§1.2 ANALYSIS

In this paper we aim to analyse the effects various algorithms, parameters and training methods have on rate and quality of learning of an artificial intelligence.

§1.3 DELIMITATIONS

This project is limited to less complex games due to the limitations of computing power and time.

§1.4 DIVISION OF WORK

Different parts of this project have been divided amongst the authors in order to make to process more effective. (More details coming)

§2. GAMES

§2.1 CONNECT FOUR

§2.2 OTHELLO

§3. DEEP LEARNING

§3.1 INTRODUCTION

§3.2 BASIC ARCHITECTURE OF NEURAL NETWORKS

§3.2.1 PERCEPTRON

§3.2.2 FULLY CONNECTED NEURAL NETWORK

§3.3 BACKPROPAGATION

§3.3.1 GRADIENT DESCENT

§3.3.2 LOSS FUNCTION

§3.4 COMPLICATIONS WITH NEURAL NETWORK TRAINING

§3.4.1 OVERFITTING

§3.4.2 VANISHING AND EXPLODING GRADIENT

§3.4.3 DIFFICULTIES IN CONVERGENCE

§3.4.4 LOCAL OPTIMA

§3.4.5 COMPUTATIONAL CHALLENGES

§3.5 TRAINING NEURAL NETWORKS

§3.5.1 BACKPROPAGATION IN DETAIL

§3.5.2 PREVENTING EXTREME GRADIENTS

§3.5.3 GRADIENT DESCENT STRATEGIES

§3.6 GENERALIZING NEURAL NETWORKS

§3.6.1 REGULARIZATION

§3.6.2 DROPOUT

§3.7 CONVOLUTIONAL NEURAL NETWORK

§3.7.1 HISTORICAL BACKGROUND

§3.7.2 ARCHITECTURE OF A CONVOLUTIONAL NETWORK

§3.7.3 TRAINING A CONVOLUTIONAL NETWORK

§3.7.4 APPLICATIONS OF CONVOLUTIONAL NETWORKS

§4. REINFORCEMENT LEARNING

§4.1 INTRODUCTION

§4.2 MONTE CARLO TREE SEARCH

§4.2.1 UPPER CONFIDENCE BOUND

§5. HEURISTIC ALGORITHMS

§5.1 MINIMAX

§5.2 ALPHABETA PRUNING

§6. METHOD

§6.1 IMPLEMENTATION

§7. RESULTS

§8. CONCLUSIONS

A. APPENDIX

A.I SOURCE CODE

<https://github.com/chopingu/hajar>

REFERENCES

- [1] Huang A Fawzi A, Balog M. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, 2022.
- [2] F Hsu M Campbell, A.J Hoane. Deep blue. *ScienceDirect*, 2002.
- [3] A. Maddison Silver, D. Huang. Mastering the game of go with deep neural networks and tree search. *Nature*, 2016.
- [4] Pritzel A Jumper J., Evans R. Highly accurate protein structure prediction with alphafold. *Nature*, 2021.
- [5] Hardesty Larry. Explained: Neural networks. *MIT News Office*, 2017.
- [6] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dhharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 2018.