
Undergraduate Topics in Computer Science

Series editor

Ian Mackie

Advisory editors

Samson Abramsky, University of Oxford, Oxford, UK

Chris Hankin, Imperial College London, London, UK

Mike Hinchey, University of Limerick, Limerick, Ireland

Dexter C. Kozen, Cornell University, Ithaca, USA

Andrew Pitts, University of Cambridge, Cambridge, UK

Hanne Riis Nielson, Technical University of Denmark, Kongens Lyngby, Denmark

Steven S. Skiena, Stony Brook University, Stony Brook, USA

Iain Stewart, University of Durham, Durham, UK


Undergraduate Topics in Computer Science (UTiCS) delivers high-quality instructional content for undergraduates studying in all areas of computing and information science. From core foundational and theoretical material to final-year topics and applications, UTiCS books take a fresh, concise, and modern approach and are ideal for self-study or for a one- or two-semester course. The texts are all authored by established experts in their fields, reviewed by an international advisory board, and contain numerous examples and problems. Many include fully worked solutions.

More information about this series at <http://www.springer.com/series/7592>

Sandro Skansi

Introduction to Deep Learning

From Logical Calculus to Artificial Intelligence

Sandro Skansi 
University of Zagreb
Zagreb
Croatia

ISSN 1863-7310 ISSN 2197-1781 (electronic)
Undergraduate Topics in Computer Science
ISBN 978-3-319-73003-5 ISBN 978-3-319-73004-2 (eBook)
<https://doi.org/10.1007/978-3-319-73004-2>

Library of Congress Control Number: 2017963994

© Springer International Publishing AG, part of Springer Nature 2018

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG part of Springer Nature
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This textbook contains no new scientific results, and my only contribution was to compile existing knowledge and explain it with my examples and intuition. I have made a great effort to cover everything with citations while maintaining a fluent exposition, but in the modern world of the ‘electron and the switch’ it is very hard to properly attribute all ideas, since there is an abundance of quality material online (and the online world became very dynamic thanks to the social media). I will do my best to correct any mistakes and omissions for the second edition, and all corrections and suggestions will be greatly appreciated.

This book uses the feminine pronoun to refer to the reader regardless of the actual gender identity. Today, we have a highly imbalanced environment when it comes to artificial intelligence, and the use of the feminine pronoun will hopefully serve to alleviate the alienation and make the female reader feel more at home while reading this book.

Throughout this book, I give historical notes on when a given idea was first discovered. I do this to credit the idea, but also to give the reader an intuitive timeline. Bear in mind that this timeline can be deceiving, since the time an idea or technique was first invented is not necessarily the time it was adopted as a technique for machine learning. This is often the case, but not always.

This book is intended to be a first introduction to deep learning. Deep learning is a special kind of learning with deep artificial neural networks, although today deep learning and artificial neural networks are considered to be the same field. Artificial neural networks are a subfield of machine learning which is in turn a subfield of both statistics and artificial intelligence (AI). Artificial neural networks are vastly more popular in artificial intelligence than in statistics. Deep learning today is not happy with just addressing a subfield of a subfield, but tries to make a run for the whole AI. An increasing number of AI fields like reasoning and planning, which were once the bastions of logical AI (also called the *Good Old-Fashioned AI*, or GOF AI), are now being tackled successfully by deep learning. In this sense, one might say that deep learning is an approach in AI, and not just a subfield of a subfield of AI.

There is an old idea from Kendo¹ which seems to find its way to the new world of cutting-edge technology. The idea is that you learn a martial art in four stages: big, strong, fast, light. ‘Big’ is the phase where all movements have to be big and correct. One here focuses on correct techniques, and one’s muscles adapt to the new movements. While doing big movements, they unconsciously start becoming strong. ‘Strong’ is the next phase, when one focuses on strong movements. We have learned how to do it correctly, and now we add strength, and subconsciously they become faster and faster. While learning ‘Fast’, we start ‘cutting corners’, and adopt a certain ‘parsimony’. This parsimony builds ‘Light’, which means ‘just enough’. In this phase, the practitioner is a master, who does everything correctly, and movements can shift from strong to fast and back to strong, and yet they seem effortless and light. This is the road to mastery of the given martial art, and to an art in general. Deep learning can be thought of an art in this metaphorical sense, since there is an element of continuous improvement. The present volume is intended not to be an all-encompassing reference, but it is intended to be the textbook for the “big” phase in deep learning. For the strong phase, we recommend [1], for the fast we recommend [2] and for the light phase, we recommend [3]. These are important works in deep learning, and a well-rounded researcher should read them all.

After this, the ‘fellow’ becomes a ‘master’ (and mastery is not the end of the road, but the true beginning), and she should be ready to tackle research papers, which are best found on [arxiv.com](https://arxiv.org/) under ‘Learning’. Most deep learning researchers are very active on [arxiv.com](https://arxiv.org/), and regularly publish their preprints. Be sure to check out also ‘Computation and Language’, ‘Sound’ and ‘Computer Vision’ categories depending on your desired specialization direction. A good practice is just to put the desired category on your web browser home screen and check it daily. Surprisingly, the [arxiv.com](https://arxiv.org/) ‘Neural and Evolutionary Computation’ is not the best place for finding deep learning papers, since it is a rather young category, and some researchers in deep learning do not tag their work with this category, but it will probably become more important as it matures.

The code in this book is Python 3, and most of the code using the library Keras is a modified version of the codes presented in [2]. Their book² offers a lot of code and some explanations with it, whereas we give a modest amount of code, rewritten to be intuitive and comment on it abundantly. The codes we offer have all been extensively tested, and we hope they are in working condition. But since this book is an introduction and we cannot assume the reader is very familiar with coding deep architectures, I will help the reader troubleshoot all the codes from this book. A complete list of bug fixes and updated codes, as well as contact details for submitting new bugs are available at the book’s repository github.com/skansi/dl_book, so please check the list and the updated version of the code before submitting a new bug fix request.

¹A Japanese martial art similar to fencing.

²This is the only book that I own two copies of, one eBook on my computer and one hard copy—it is simply that good and useful.

Artificial intelligence as a discipline can be considered to be a sort of ‘philosophical engineering’. What I mean by this is that AI is a process of taking philosophical ideas and making algorithms that implement them. The term ‘philosophical’ is taken broadly as a term which also encompasses the sciences which recently³ became independent sciences (psychology, cognitive science and structural linguistics), as well as sciences that are hoping to become independent (logic and ontology⁴).

Why is philosophy in this broad sense so interesting to replicate? If you consider what topics are interesting in AI, you will discover that AI, at the most basic level, wishes to replicate philosophical concepts, e.g. to build machines that can think, know stuff, understand meaning, act rationally, cope with uncertainty, collaborate to achieve a goal, handle and talk about objects. You will rarely see a definition of an AI agent using non-philosophical terms such as ‘a machine that can route internet traffic’, or ‘a program that will predict the optimal load for a robotic arm’ or ‘a program that identifies computer malware’ or ‘an application that generates a formal proof for a theorem’ or ‘a machine that can win in chess’ or ‘a subroutine which can recognize letters from a scanned page’. The weird thing is, all of these are actual historical AI applications, and machines such as these always made the headlines.

But the problem is, once we got it to work, it was no longer considered ‘intelligent’, but merely an elaborate computation. AI history is full of such examples.⁵ The systematic solution of a certain problem requires a full formal specification of the given problem, and after a full specification is made, and a known tool is applied to it,⁶ it stops being considered a mystical human-like machine and starts being considered ‘mere computation’. Philosophy deals with concepts that are inherently tricky to define such as knowledge, meaning, reference, reasoning, and all of them are considered to be essential for intelligent behaviour. This is why, in a broad sense, AI is the engineering of philosophical concepts.

But do not underestimate the engineering part. While philosophy is very prone to reexamining ideas, engineering is very progressive, and once a problem is solved, it is considered done. AI has the tendency to revisit old tasks and old problems (and this makes it very similar to philosophy), but it does require measurable progress, in the sense that new techniques have to bring something new (and this is its

³Philosophy is an old discipline, dating back at least 2300 years, and ‘recently’ here means ‘in the last 100 years’.

⁴Logic, as a science, was considered independent (from philosophy and mathematics) by a large group of logicians for at least since Willard Van Orman Quine’s lectures from the 1960s, but thinking of ontology as an independent discipline is a relatively new idea, and as far as I was able to pinpoint it, this intriguing and promising initiative came from professor Barry Smith from the Department of Philosophy of the University of Buffalo.

⁵John McCarthy was amused by this phenomenon and called it the ‘look ma’, no hands’ period of AI history, but the same theme keeps recurring.

⁶Since new tools are presented as new tools for existing problems, it is not very common to tackle a new problem with newly invented tools.

engineering side). This novelty can be better results than the last result on that problem,⁷ the formulation of a new problem⁸ or results below the benchmark but which can be generalized to other problems as well.

Engineering is progressive, and once something is made, it is used and built upon. This means that we do not have to re-implement everything anew—there is no use in reinventing the wheel. But there is value to be gained in understanding the idea behind the invention of the wheel and in trying to make a wheel by yourself. In this sense, you should try to recreate the codes we will be exploring, and see how they work and even try to re-implement a completed Keras layer in plain Python. It is quite probable that if you manage your solution will be considerably slower, but you will have gained insight. When you feel you understand it as much as you would like, you should just use Keras or any other framework as building bricks to go on and build more elaborate things.

In today's world, everything worth doing is a team effort and every job is then divided in parts. My part of the job is to get the reader started in deep learning. I would be proud if a reader would digest this volume, put it on a shelf, become and active deep learning researcher and never consult this book again. To me, this would mean that she has learned everything there was in this book and this would entail that my part of the job of getting one started⁹ in deep learning was done well. In philosophy, this idea is known as Wittgenstein's ladder, and it is an important practical idea that will supposedly help you in your personal exploration—exploitation balance.

I have also placed a few Easter eggs in this volume, mainly as unusual names in examples. I hope that they will make reading more lively and enjoyable. For all who wish to know, the name of the dog in Chap. 3 is Gabi, and at the time of publishing, she will be 4 years old. This book is written in plural, following the old academic custom of using *pluralis modestiae*, and hence after this preface I will no longer use the singular personal pronoun, until the very last section of the book.

I would wish to thank everyone who has participated in any way and made this book possible. In particular, I would like to thank Siniša Urošev, who provided valuable comments and corrections of the mathematical aspects of the book, and to Antonio Šajatović, who provided valuable comments and suggestions regarding memory-based models. Special thanks go to my wife Ivana for all the support she gave me. I hold myself (and myself alone) responsible for any omissions or mistakes, and I would greatly appreciate all feedback from readers.

Zagreb, Croatia

Sandro Skansi

⁷This is called the *benchmark* for a given problem, it is something you must surpass.

⁸Usually in the form of a new dataset constructed from a controlled version of a philosophical problem or set of problems. We will have an example of this in the later chapters when we will address the bAbI dataset.

⁹Or, perhaps, 'getting initiated' would be a better term—it depends on how fond will you become of deep learning.

References

1. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT press, Cambridge, 2016)
2. A. Gulli, S. Pal, *Deep Learning with Keras* (Packt publishing, Birmingham, 2017)
3. G. Montavon, G. Orr, K.R. Muller, *Neural Networks: Tricks of the Trade* (Springer, New York, 2012)

Contents

1	From Logic to Cognitive Science	1
1.1	The Beginnings of Artificial Neural Networks	1
1.2	The XOR Problem	5
1.3	From Cognitive Science to Deep Learning	8
1.4	Neural Networks in the General AI Landscape	11
1.5	Philosophical and Cognitive Aspects	12
	References	15
2	Mathematical and Computational Prerequisites	17
2.1	Derivations and Function Minimization	17
2.2	Vectors, Matrices and Linear Programming	25
2.3	Probability Distributions	32
2.4	Logic and Turing Machines	39
2.5	Writing Python Code	41
2.6	A Brief Overview of Python Programming	43
	References	49
3	Machine Learning Basics	51
3.1	Elementary Classification Problem	51
3.2	Evaluating Classification Results	57
3.3	A Simple Classifier: Naive Bayes	59
3.4	A Simple Neural Network: Logistic Regression	61
3.5	Introducing the MNIST Dataset	68
3.6	Learning Without Labels: K-Means	70
3.7	Learning Different Representations: PCA	72
3.8	Learning Language: The Bag of Words Representation	75
	References	77
4	Feedforward Neural Networks	79
4.1	Basic Concepts and Terminology for Neural Networks	79
4.2	Representing Network Components with Vectors and Matrices	82
4.3	The Perceptron Rule	84
4.4	The Delta Rule	87

4.5	From the Logistic Neuron to Backpropagation	89
4.6	Backpropagation	93
4.7	A Complete Feedforward Neural Network	102
	References	105
5	Modifications and Extensions to a Feed-Forward Neural Network	107
5.1	The Idea of Regularization	107
5.2	L_1 and L_2 Regularization	109
5.3	Learning Rate, Momentum and Dropout	111
5.4	Stochastic Gradient Descent and Online Learning	116
5.5	Problems for Multiple Hidden Layers: Vanishing and Exploding Gradients	118
	References	119
6	Convolutional Neural Networks	121
6.1	A Third Visit to Logistic Regression	121
6.2	Feature Maps and Pooling	125
6.3	A Complete Convolutional Network	127
6.4	Using a Convolutional Network to Classify Text	130
	References	132
7	Recurrent Neural Networks	135
7.1	Sequences of Unequal Length	135
7.2	The Three Settings of Learning with Recurrent Neural Networks	136
7.3	Adding Feedback Loops and Unfolding a Neural Network	139
7.4	Elman Networks	140
7.5	Long Short-Term Memory	142
7.6	Using a Recurrent Neural Network for Predicting Following Words	145
	References	152
8	Autoencoders	153
8.1	Learning Representations	153
8.2	Different Autoencoder Architectures	156
8.3	Stacking Autoencoders	158
8.4	Recreating the Cat Paper	161
	References	163
9	Neural Language Models	165
9.1	Word Embeddings and Word Analogies	165
9.2	CBOW and Word2vec	166
9.3	Word2vec in Code	168

9.4	Walking Through the Word-Space: An Idea That Has Eluded Symbolic AI	171
	References.	173
10	An Overview of Different Neural Network Architectures.	175
10.1	Energy-Based Models.	175
10.2	Memory-Based Models.	178
10.3	The Kernel of General Connectionist Intelligence: The bAbI Dataset	181
	References.	182
11	Conclusion	185
11.1	An Incomplete Overview of Open Research Questions	185
11.2	The Spirit of Connectionism and Philosophical Ties	186
	Reference	187
	Index	189