CHAPTER TWO: HISTORICAL COMMENTS

The introduction of the term "deep-learning" in 2006 by psychologist Geoffrey Hinton, heralded a new age in artificial intelligence leading to a ~~that would sire an enormous amount~~ swarm of written works by journalists and research groups heralding a departure from "traditional" computing [20, 23]. Much of what would then transpire in technical literature and in the mainstream dealt with the practical details of the research, the highly imaginative research applications come out of engineering teams, such as that which would took place following IBM Watson's ~~liveried~~ brief stint on the Jeopardy game show and later media around the Google Brain project through 2011 to 2012, when artificial intelligence researchers themselves would then frequently encounter concerns regarding academic integrity [15]. Many depictions ~~works~~ of ~~on this~~ deep-learning or the "deep machine-learning" age ~~have~~ described the massive gains in both predictive power and scientific advancements that would be achieved through a new entrepreneurial culture manned by engineers themselves - at the time, a dawning shock. In most cases, they ~~have~~ described the fantastic skills of data-scientists who would develop or reimagine General AI (GAI) - the kind usually touched upon in science-fiction or innovation as a result of deep-learning advancements; where reality would be allowed to suffuse depictions found in folklore. Other critical observers decried potentially high level concerns of information security and privacy, which would exist after this new age began; with governments seemingly failing to account for policy gaps standing-by as corporations ~~continued to~~ poured billions into deploying and supporting these new deep-learning systems as they came online [21, 32]. Some authors sought to question the credibility of the underlying data sets proclaiming that human biases had made their way into training sets for large-scale artificial intelligence systems in sensitive application areas; for example in employment candidate screening, where it was thought any mass adoption of machine-led decision making systems would seemingly encourage discrimination[1].

Pattern Recognition: Foundational Basis for Algorithmic Modelling and Understanding in AI

Certainly, anybody so wishing to attempt to write a book on the use, design and implementation of AI and machine learning, against this backdrop, must ~~ask~~ quickly decide where to kick-off. Texts on AI and machine learning ~~have~~ often at times start with a discussion on McCulluch and Pitts developments in neural network models of the 1940's [24] or may commence with the efforts 1957 with Rosenblatt's perceptron[31]. However this book is considerably less zealous. An introductory text, it commences with pattern-recognition, opted for because we ascribe to the view that an understanding the fundamental principles of pattern recognition provides a more accessible and practical foundation for those new to the field of AI, thereby promoting a smoother transition later into more complex

[1]This is an old problem; as far back as 1988 the UK Commission for Racial Equality cautioned an elite British medical school for admission contretemps caused by use of a computer algorithm that discriminated in the admissions process. The algorithm which would match human admissions decisions at an accuracy of 90 to 95 percent, was however found to be biased against females and applicants with non-European sounding names. The medical school in it's response explained it actually had a relatively high proportion of students with non-European names when compared to other similar institutions - therefore demanding the use of such systems in sensitive areas be immediately halted[10, 27, 11].

notions and methodologies. Understanding the foundational concepts of pattern recognition is pivotal in fully comprehending the progression and implications of deep learning.

The starting point for theory-based modelling of learning algorithms will usually require a specification of the underlying statistical or mathematical model that captures the essential characteristics of the data, including the assumptions and any relationships that guide the underlying process. Pattern recognition is often viewed as a direct means for recognizing patterns and irregularities of data sets. Thus raw data upon which a pattern recognition model has been well fit, should allow data to be classified on an unambiguous basis. The view of pattern recognition as just another machine learning technique is at best a crude oversimplification, in that it ignores the nuances in methodology, focus, and the applications that clearly distinguish it from other machine learning approaches; particularly in terms of the relative gain or loss in performance. Nevertheless, pattern recognition models ~~are a good starting~~ serve as our starting point for a broader analysis of AI and of machine learning[2]. This book believes that in order to truly grasp the nature of AI's evolution and the current deep-learning revolution, one must first understand the basics in how machines learn. ~~from patterns~~. This approach is designed to illuminate not just the capabilities and potential of learning algorithms, but highlight their complexities, challenges, the ethical and societal questions they provoke.

~~Understanding pattern recognition also serves as a starting point, this book then aims to provide a comprehensive, insightful journey through the evolution of AI and machine learning, underscoring the critical role of deep learning in these processes.~~

The Evolution of Parallel Processing: Pioneering the Computational Revolution in AI and Deep Learning.

Advancements in parallel processing architectures, as highlighted by John L. Hennessy and David A. Patterson, led to an increased focus on parallel programming particularly the inherent design complexities [19]. By utilizing multiple processing elements, such as cores or processors to execute tasks concurrently, parallel processing appears to pave the way for the considerable development in parallel programs, resulting in significant performance improvements in terms of throughput and the number of threads in the latter. In retrospect, parallel processing literature reveals a more complex picture, indicating that further performance gains through increasing clock frequency faced a limitations necessitating a shift towards more sophisticated approaches around 2003, due to concerns over energy consumption and heat dissipation [19, 8, 2].

Massively-parallel processing fundamentally altered the landscape of AI and deep learning by enabling the com-

[2]Pattern recognition algorithms, as a precursor to advanced machine learning, hold the key to understanding the foundational concepts on how machines perceive, interpret, and interact in the world. Pattern recognition offers an additional distinct advantage in that it often simplifies the complex task of identifying features and processing large data sets by using traditional mathematical techniques, thereby often enabling a more rapid and targeted analysis than other learning techniques.

putation and processing of large volumes of data simultaneously, to ensure performance gains [13]. As a result, the ability to handle and learn from larger datasets became possible, paving the way for the rise of complex neural network architectures and more sophisticated learning algorithms in next-generation algorithms [5]. The evolution in processing power, which allowed for the implementation of algorithms characterized by increased computational intricacy, higher-dimensional parameter spaces, and better mathematically-defined interconnections between computational units or nodes, accordingly is the focus of the first part of this text and is covered in brisk detail. This section of the book delves into the technical underpinnings of massively-parallel processing, its role in propelling the field of deep learning forward, and its continued relevance to contemporary AI researchers. We will continue dissect key developments, innovative applications, and the technical challenges faced in harnessing the full potential of this processing power. As we embark on this exploration, we aim to provide a comprehensive understanding of how massively-parallel processing has been pivotal in AI and deep learnings continued transformative evolution.

The Democratization of Parallel Computing: GPUs and the Concurrency Revolution.

Despite the throughput gains acquired in the practice of parallel programming, parallel programs were almost always limited to large and often expensive computers, leading to limited developer adoption. Development costs are ~~of software development is~~ are a significant cost component, ~~applications that were~~ when applications are developed for processors with a small market footprint, referred to as its *installed base*, often times this leads to halted projects [30, 3, 33]. Therefore, historically only a limited number of computer programs could be successfully funded on traditional parallel computing platforms, with even lower success probabilities [16]. A lack of standardisation of Floating-Point Arithmetic formats across various processors would also limit the portability of computer programs compared to CPUs, again adding to the cost of development even for more experienced engineers [4]; basically development on CPUs is more cost-effective and efficient [17, 28].

What is often available to engineering users as the most robust and therefore the most important form of massively-parallel processors is the graphical processing unit (GPU). It was in this respect that development of the GPU is also compelling for the hardware engineering and scientific communities ~~history of computer programming~~. Therefore GPU's came to ~~have begun however to~~ provide a viable mass-market alternative to the otherwise "prohibitively specialized architectures" markets given to their proliferation. As a direct consequence of these technological advancements, GPUs have revolutionized the accessibility of parallel computing, bringing it within the reach of a broader spectrum of developers and applications; this has led to an increase in adoption and utilization attributable to enhanced clarity.

This democratization of parallel computing technology has facilitated an era of unprecedented innovation and exploration in various fields including machine learning, real-time graphics and industrial-scale scientific research.

Furthermore, convergence of GPU architecture with general-purpose computing standards has truly cemented their role as a mainstream, rather than a niche computing platform. For example numerical computing approximations of arithmetic in more recent iterations of GPUs are now comparable to that of CPUs, due to their implementation of the IEEE Standard for Floating-Point Arithmetic (IEEE 754) [3] [22, 17]. Especially true in heterogeneous computing environments, where floating point accuracy is important to achieve the highest performance as operations will be performed on different types of hardware. [4]. ~~This paves the way for a smoother transition of existing applications from CPUs.~~

The advent of the mass-market, low form factor GPU provided significant impetus to massively-parallel processing, colloquially referred to as the *concurrency revolution* [reference] highlighting how this paradigm has lead to a shift from traditional single thread platforms [26, 1]. This brings standardisation also to the market place and bolsters the swift trajectory of advancements in floating-point performance distinctly observable since 2003, a period before which parallel platforms attracted a considerable economic premium [29]. The foundational assertion of this text is that GPUs by virtue of an inherently parallel design and their high-throughput orientation, serve as a significant driving force behind advancements in parallel programming realm.

Purpose and Scope.

Our text, *Learning and Pattern Recognition for Programmable Systems* was born out of the conviction that a grasp of computer programming paradigms is essential to an engineer's competence. Without a framework presented through literature, engineering theory often drifts into treatises on computational, mathematical, or even physics theory. It is in this sense that *Learning and Pattern Recognition for Programmable Systems* seeks to present a text on the design of machine-learning algorithms in the deep-learning age. Other attempts to give academically robust texts on massive parallel-programming for pattern recognition and machine learning have been too abstract or were strategically misaligned.

Some readers will be dissatisfied however, that their favourite paradigms or strategies in machine learning are not included. To write about the implementation of all the possible machine learning paradigms and strategies on a massively-parallel computing platform would be inconceivable. Any attempt therefore to present such a text of that nature would at best be anecdotal, overly abridged, or lacking in appeal. This narrative therefore attempts to present the subject matter in a linear, rather accepted first-hand form. ~~We assume that the reviewer has at least basic experience in C programming.~~

As Christopher Bishop wrote at the start of his seminal text, Learning and Pattern Recognition for Pro-

[3] When high-performance computing works with inexact results, their hardware decisions affect accuracy particularly where resources are shared

[4] The additional memory and bandwidth bottleneck inherent in GPUs, has lead to a preference for 32-bit operations and more recently even 16-bit being more desirable [25]

grammable Systems this book aims to lay down the theoretical foundations of massively-parallel processing and heterogeneous parallel computing, as key strategies in pattern recognition and machine learning, "grounded in the principles of engineering"[6]. This book explains engineering fundamentals, as one might come across in an introductory computational engineering text or a literature review, for those who wish to do doing state-of-the-art work in the field; it is not a treatise on the high-performance computing revolution. This book adopts a fundamentals based approach to understanding learning algorithm development and consequently provides an implementation guide for deep-learning practitioners. The "deep-learning age" has naturally evoked many discussions on the impact of learning algorithms ~~and predictability,~~ programmable processors, and even the impact these tools will have on the evolution of the human race by both practitioners and academics [18, 9, 14]. Individuals interested in a broader historical perspective on the evolution of deep-learning since 2006 have also touched on the limitations of AI; for example, autonomous agents and their integration into everyday environments like urban transportation with self-driving cars or healthcare through personalised medicine [7, 34, 12]. These type of applications continue to face overwhelming odds. Mostly due to my desire to understand these limitations, I commenced research for this text and thereby better understand the nuances between the many different algorithm choices. To that goal we dedicate the following book. *Learning and Pattern Recognition for Programmable Systems* aims not to be prescriptive but rather a composition for practitioners about the use, design and implementation of deep-learning from its pattern recognition roots through to the development of machine learning.

References

[1] Shogo Asano, Tsutomu Maruyama, and Yoshiki Yamaguchi. Performance comparison of fpga, gpu and cpu in image processing. *2009 International Conference on Field Programmable Logic and Applications*, pages 126–131, 2009.

[2] Krste Asanovic, Ras Bodik, Bryan Christopher Catanzaro, Joseph James Gebis, Parry Husbands, Kurt Keutzer, David A. Patterson, William Lester Plishker, John Shalf, Samuel Webb Williams, and Katherine A. Yelick. The landscape of parallel computing research: A view from berkeley. Technical report, Technical Report No. UCB/EECS-2006-183, EECS Department, University of California, Berkeley, 2006.

[3] William Aspray. A historical reflection on the future. *Annals of the History of Computing, IEEE*, 18(2):48–56, 1996.

[4] David Bailey and Eric Barszcz. High-performance floating-point arithmetic on fpgas. In *International Conference on Application-Specific Systems, Architectures and Processors*, pages 155–166. IEEE, 1996.

[5] Tal Ben-Nun and Torsten Hoefler. Demystifying parallel and distributed deep learning: An in-depth concurrency analysis. *ACM Computing Surveys (CSUR)*, 52(4):1–43, 2019.

[6] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[7] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

[8] Shekhar Borkar and Andrew A. Chien. Thousand core chips: A technology perspective. In *Proceedings of the 44th Annual Design Automation Conference*, pages 746–749. ACM, 2007.

[9] Nick Bostrom. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press, 2014.

[10] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. *Proceedings of Machine Learning Research*, 81:1–15, 2018.

[11] Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, 2017.

[12] Jonathan H Chen and Steven M Asch. Machine learning and prediction in medicine—beyond the peak of inflated expectations. *New England Journal of Medicine*, 376(26):2507–2509, 2017.

[13] Trishul Chilimbi, Yutaka Suzue, Johnson Apacible, and Karthik Kalyanaraman. Project adam: Building an efficient and scalable deep learning training system. In *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14)*, pages 571–582, 2014.

[14] Pedro Domingos. *The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World*. Basic Books, 2015.

[15] David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A Kalyanpur, Adam Lally, J William Murdock, Eric Nyberg, John Prager, et al. Building watson: An overview of the deepqa project. *AI magazine*, 31(3):59–79, 2010.

[16] Ian Foster. Designing and building parallel programs: Concepts and tools for parallel software engineering. *Addison-Wesley*, 1995.

[17] David Goldberg. Every computer floating-point arithmetic should implement. In *Proceedings of the 10th symposium on Computer arithmetic*, pages 2–14. IEEE Computer Society Press, 1991.

[18] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[19] John L. Hennessy and David A. Patterson. *Computer Architecture: A Quantitative Approach*. Morgan Kaufmann, 6 edition, 2017.

[20] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[21] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. Deep models under the gan: Information leakage from collaborative deep learning. *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 603–618, 2017.

[22] W. Kahan. *Branch Cuts for Complex Elementary Functions, or Much Ado About Nothing's Sign Bit*. Springer, 1987.

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[24] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[25] Paulius Micikevicius. Mixed precision training. In *Proceedings of the International Conference on Learning Representations (ICLR)*, volume 2018, 2018.

[26] John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. Gpu computing. volume 96, pages 879–899. IEEE, 2010.

[27] Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453, 2019.

[28] Michael L Overton. Numerical computing with ieee floating-point arithmetic: Including one theorem, one rule of thumb, and one hundred and one exercises. Technical report, 1997.

[29] John D Owens, David Luebke, Naga Govindaraju, Mark Harris, Jens Krüger, Aaron E Lefohn, and Timothy J Purcell. A survey of general-purpose computation on graphics hardware. volume 26, pages 80–113. Wiley Online Library, 2007.

[30] Peter Pacheco. *An Introduction to Parallel Programming*. Elsevier, 2011.

[31] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[32] Reza Shokri, Marco Stronati, Cong Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.

[33] Steven Swanson, Ken Michelson, and Michael Oskin. Increasing the instruction fetch rate via multiple branch prediction and a branch address cache. *ACM SIGARCH Computer Architecture News*, 35(1):241–252, 2007.

[34] Eric J Topol. High-performance medicine: the convergence of human and artificial intelligence. *Nature Medicine*, 25(1):44–56, 2019.