Introduction

There are three main ingredients to this thesis:

- quantum computing,
- natural language processing,
- and category theory.

We will introduce them in order, followed by a summary of the thesis.

What are quantum computers good for?

> Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy.
>
> *Simulating Physics with Computers*, Feynman (1981)

Quantum computers harness the principles of quantum theory such as superposition and entanglement to solve information-processing tasks. In the last 42 years, quantum computing has gone from theoretical speculations to the implementation of machines that can solve problems beyond what is possible with classical means. This section will sketch a brief history of the field, and of its future challenges.

In 1980, Benioff [Ben80] takes the abstract definition of a computer and makes it physical: he designs a quantum mechanical system whose time evolution encodes the computation steps of a given Turing machine. In retrospective, this may be taken as the first proof that quantum mechanics can simulate classical computers. The same year, Manin [Man80] looks at the opposite direction: he argues that it would take exponential time for a classical computer to simulate a generic quantum system. In the following years, Feynman [Fey82; Fey85] comes to the same conclusion and suggests a way to simulate quantum mechanics much more efficiently: building a quantum computer!

So what are quantum computers good for? Feynman's intuition gives us a first, trivial answer: at least quantum computers can simulate quantum mechanics efficiently. Deutsch [Deu85] makes the question formal by defining quantum Turing

machines and quantum circuits. Deutsch and Jozsa [DJ92] design the first quantum algorithm and prove that it solves *some* problem exponentially faster than any classical *deterministic*[1] algorithm. Simon [Sim94] improves on their result by designing a problem that a quantum computer can solve exponentially faster than any classical algorithm. Deutsch-Jozsa and Simon relied on oracles[2] and promises[3] and their problems have little practical use. However, they inspired Shor's algorithm [Sho94] for prime factorisation and discrete logarithm. These two problems are believed to require exponential time for a classical computer and their hardness is at the basis of the public-key cryptography schemes currently used on the internet.

In 1997, Grover provides another application for quantum computers: "searching for a needle in a haystack" [Gro97]. Formally, given a black box function $f : X \to \{0, 1\}$ and the promise that there is a unique $x \in X$ with $f(x) = 1$, Grover's algorithm finds x in $O(\sqrt{|X|})$ steps, quadratically faster than the optimal $O(|X|)$ classical algorithm. Grover's algorithm may be used to brute-force symmetric cryptographic keys twice bigger than what is possible classically [BBD09]. It can also be used to obtain quadratic speedups for the exhaustive search involved in the solution of NP-hard problems such as constraint satisfaction [Amb04]. Independently, Benett et al. [Ben+97] prove that Grover's algorithm is in fact optimal, adding evidence to the conjecture that quantum computers cannot solve these NP-hard problems in polynomial time. The following year, Chuang et al. [CGK98] give the first experimental demonstration of a quantum algorithm, running Grover's algorithm on two qubits.

Shor's and Grover's discovery of the first real-world applications sparked a considerable interest in quantum computing. The core of these two algorithms has then been abstracted away in terms of two subroutines: phase estimation [Kit95] and amplitude amplification [Bra+02], respectively. Making use of both these subroutines, the HHL[4] algorithm [HHL09] tackles one of the most ubiquitous problems in scientific computing: solving systems of linear equations. Given a matrix $A \in \mathbb{R}^{n \times n}$ and a vector $\vec{b} \in \mathbb{R}^n$, we want to find a vector $\vec{x}$ such that $A\vec{x} = \vec{b}$. Under some assumptions on the sparsity and the condition number of A, HHL finds (an approximation of) x in time logarithmic in n when a classical algorithm would take quadratic time simply to read the entries of A. This initiated a new wave of enthusiasm for quantum computing with the promise of exponential speedups for machine learning tasks such as regression [WBL12], clustering [LMR13], classification [RML14], dimensionality reduction [LMR14] and recommendation [KP16]. The narrative is appealing: machine learning is about finding patterns in large amounts of data represented as high-dimensional vectors and tensors, which is precisely what quantum computers are good at.

However, the exponential speedup of HHL comes with some caveats, thoroughly analysed by Aaronson [Aar15]. Two of these challenges are common to many quantum algorithms: 1) the efficient encoding of classical data into quantum

[1] A classical *randomised* algorithm solves the problem in constant time with high probability.
[2] An oracle is a black box that allows a Turing machine to solve a certain problem in one step.
[3] The input is promised to satisfy a certain property, which may be hard to check.
[4] Named after its discoverers Harrow, Hassidim and Lloyd.

states and 2) the efficient extraction of classical data via quantum measurements. Indeed, what HHL really takes as input is not a vector $\vec{b}$ but a quantum state $|b\rangle = \sum_{i=1}^{n} b_i|i\rangle$. Either a) the input vector $\vec{b}$ has enough structure that we can describe it with a simple, explicit formula. This is the case for example in the calculation of electromagnetic scattering cross-sections [CJS13]. Or b) we need to assume that our classical data has been loaded onto a quantum random-access memory (qRAM) [GLM08]. Not only is qRAM a daunting challenge from an engineering point of view, in some cases it also requires too much error correction for the state preparation to be efficient [Aru+15]. Symmetrically, the output of HHL is not the solution vector $\vec{x}$ itself but a quantum state $|x\rangle$ from which we can measure some observable $\langle x|M|x\rangle$. If preparing the state $|b\rangle$ requires a number of gates exponential in the number of qubits, or if we need exponentially many measurements of $|x\rangle$ to compute our classical output, then the quantum speedup disappears.

Shor, Grover and HHL all assume *fault-tolerant* quantum computers [Sho96]. Indeed, any machine we can build will be subject to noise when performing quantum operations, errors are inevitable. We need an error correcting code that can correct these errors faster than they appear. This is the content of the *quantum threshold theorem* [AB08] which proves the possibility of fault-tolerant quantum computing given physical error rates below a certain threshold. One noteworthy example of such a quantum error correction scheme is Kitaev's toric code [Kit03] and the general idea of topological quantum computation [Fre+03] which offers the long-term hope for a quantum computer that is fault-tolerant "by its physical nature". However this hope relies on the existence of quasi-particles called Majorana zero-modes, which as of 2021 has yet to be experimentally demonstrated [Bal21].

The road to large-scale fault-tolerant quantum computing will most likely be a long one. So in the meantime, what can we do with the noisy intermediate-scale quantum machines we have available today, in the so-called NISQ era [Pre18]? Most answers involve a hybrid classical-quantum approach where a classical algorithm is used to optimise the preparation of quantum states [McC+16]. Prominent examples include the quantum approximate optimisation algorithm (QAOA [FGG14]) for combinatorial problems such as maximum cut and the variational quantum eigensolver (VQE [Per+14]) for approximating the ground state of chemical systems. These variational algorithms depend on the choice of a parameterised quantum circuit called the *ansatz*, based on the structure of the problem and the resources available. Some families of ansätze such as instantaneous quantum polynomial-time (IQP) circuits are believed to be hard to simulate classically even at constant depth [SB09], opening the door to potentially near-term NISQ speedups.

Although the hybrid approach first appeared in the context of machine learning [Ban+08], the idea of using parameterised quantum circuits as machine learning models went mostly unnoticed for a decade [BLS19]. It was rediscovered under the name of quantum neural networks [FN18] then implemented on two-qubits [Hav+19], generating a new wave of attention for quantum machine learning. One of the many challenges on the way to solving real-world problems with parameterised quantum circuits is the existence of *barren plateaus* [McC+18]: with a random circuit as ansatz, the probability of non-zero gradients is exponentially small in

the number of qubits and our classical optimisation gets lost in a flat landscape. One can help but notice the striking similarity with the vanishing gradient problem for classical neural networks, formulated twenty years earlier [Hoc98]. Barren plateaus disappear when circuits have enough structure, as in the case of quantum convolutional networks [Pes+21], they can also be mitigated by structured initialisation strategies [Gra+19].

Random quantum circuits may be unsuitable for machine learning, but they play a crucial role in the quest for *quantum advantage*, the experimental demonstration of a quantum computer solving a task that cannot be solved by classical means in any reasonable time. We are back to Feynman's original intuition: simulating a random quantum circuit is the perfect candidate for such a task. By construction it is an easy task for a quantum computer (given that we can mitigate the effect of noise). If a classical computer could solve it efficiently, then the polynomial hierarchy would collapse to the third level, which is almost as unlikely as P=NP.

Google were the first to claim such an advantage with their 53-qubit Sycamore computer [Aru+19]. Their claim was almost immediately contested by IBM's classical simulation of 54 qubits [Ped+19]. The following year, USTC makes a new claim [Zho+20] based on the hardness of boson sampling [AA11]. The claim of quantum advantage has an asymptotic nature, thus its definite proof will span a long period of time.

The demonstration of a quantum advantage for a useful task is still open.

References

[Aar15] Scott Aaronson. “Read the Fine Print”. In: *Nature Physics* 11.4 (Apr. 2015), pp. 291–293.

[AA11] Scott Aaronson and Alex Arkhipov. “The Computational Complexity of Linear Optics”. In: *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*. 2011, pp. 333–342.

[AB08] Dorit Aharonov and Michael Ben-Or. “Fault-Tolerant Quantum Computation with Constant Error Rate”. In: *SIAM Journal on Computing* 38.4 (Jan. 2008), pp. 1207–1282.

[Amb04] A. Ambainis. “Quantum Search Algorithms”. In: *ACM SIGACT News* 35.2 (June 2004), pp. 22–35.

[Aru+15] Srinivasan Arunachalam, Vlad Gheorghiu, Tomas Jochym-O’Connor, Michele Mosca, and Priyaa Varshinee Srinivasan. “On the Robustness of Bucket Brigade Quantum RAM”. In: *New Journal of Physics* 17.12 (Dec. 2015), p. 123010.

[Aru+19] Frank Arute et al. “Quantum Supremacy Using a Programmable Superconducting Processor”. In: *Nature* 574.7779 (Oct. 2019), pp. 505–510.

[Bal21] Philip Ball. *Major Quantum Computing Strategy Suffers Serious Setbacks*. https://www.quantamagazine.org/major-quantum-computing-strategy-suffers-serious-setbacks-20210929/. Sept. 2021.

[Ban+08] Jeongho Bang, James Lim, M. S. Kim, and Jinhyoung Lee. “Quantum Learning Machine”. In: *arXiv:0803.2976 [quant-ph]* (Mar. 2008). arXiv: `0803.2976 [quant-ph]`.

[BLS19] Marcello Benedetti, Erika Lloyd, and Stefan Sack. “Parameterized Quantum Circuits as Machine Learning Models”. In: *arXiv:1906.07682 [quant-ph]* (June 2019). arXiv: `1906.07682 [quant-ph]`.

[Ben80] Paul Benioff. “The Computer as a Physical System: A Microscopic Quantum Mechanical Hamiltonian Model of Computers as Represented by Turing Machines”. In: *Journal of Statistical Physics* 22.5 (May 1980), pp. 563–591.

[Ben+97] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. “Strengths and Weaknesses of Quantum Computing”. In: *SIAM Journal on Computing* 26.5 (Oct. 1997), pp. 1510–1523.

[BBD09] Daniel J Bernstein, Johannes Buchmann, and Erik Dahmén. *Post-Quantum Cryptography*. Berlin: Springer, 2009.

[Bra+02] Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. "Quantum Amplitude Amplification and Estimation". In: *arXiv:quant-ph/0005055* 305 (2002), pp. 53–74. arXiv: quant-ph/0005055.

[CGK98] Isaac L. Chuang, Neil Gershenfeld, and Mark Kubinec. "Experimental Implementation of Fast Quantum Searching". In: *Physical Review Letters* 80.15 (Apr. 1998), pp. 3408–3411.

[CJS13] B. D. Clader, B. C. Jacobs, and C. R. Sprouse. "Preconditioned Quantum Linear System Algorithm". In: *Physical Review Letters* 110.25 (June 2013), p. 250504.

[Deu85] David Deutsch. "Quantum Theory, the Church–Turing Principle and the Universal Quantum Computer". In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 400.1818 (1985), pp. 97–117.

[DJ92] David Deutsch and Richard Jozsa. "Rapid Solution of Problems by Quantum Computation". In: *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences* 439.1907 (Dec. 1992), pp. 553–558.

[FGG14] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. "A Quantum Approximate Optimization Algorithm". In: *arXiv:1411.4028 [quant-ph]* (Nov. 2014). arXiv: 1411.4028 [quant-ph].

[FN18] Edward Farhi and Hartmut Neven. "Classification with Quantum Neural Networks on Near Term Processors". In: *arXiv:1802.06002 [quant-ph]* (Aug. 2018). arXiv: 1802.06002 [quant-ph].

[Fey82] Richard P. Feynman. "Simulating Physics with Computers". In: *International Journal of Theoretical Physics* 21.6 (June 1982), pp. 467–488.

[Fey85] Richard P Feynman. "Quantum Mechanical Computers". In: *Optics news* 11.2 (1985), pp. 11–20.

[Fre+03] Michael Freedman, Alexei Kitaev, Michael Larsen, and Zhenghan Wang. "Topological Quantum Computation". In: *Bulletin of the American Mathematical Society* 40.1 (2003), pp. 31–38.

[GLM08] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. "Quantum Random Access Memory". In: *Physical Review Letters* 100.16 (Apr. 2008), p. 160501.

[Gra+19] Edward Grant, Leonard Wossnig, Mateusz Ostaszewski, and Marcello Benedetti. "An Initialization Strategy for Addressing Barren Plateaus in Parametrized Quantum Circuits". In: *Quantum* 3 (Dec. 2019), p. 214.

[Gro97] Lov K. Grover. "Quantum Mechanics Helps in Searching for a Needle in a Haystack". In: *Physical Review Letters* 79.2 (July 1997), pp. 325–328. arXiv: quant-ph/9706033.

[HHL09] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. "Quantum Algorithm for Linear Systems of Equations". In: *Physical Review Letters* 103.15 (Oct. 2009), p. 150502.

[Hav+19] Vojtech Havlicek et al. "Supervised Learning with Quantum Enhanced Feature Spaces". In: *Nature* 567.7747 (Mar. 2019), pp. 209–212. arXiv: 1804.11326.

[Hoc98] Sepp Hochreiter. "The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 06.02 (Apr. 1998), pp. 107–116.

[KP16] Iordanis Kerenidis and Anupam Prakash. "Quantum Recommendation Systems". In: *arXiv:1603.08675 [quant-ph]* (Mar. 2016). arXiv: 1603.08675 [quant-ph].

[Kit95] A. Yu Kitaev. "Quantum Measurements and the Abelian Stabilizer Problem". In: *arXiv:quant-ph/9511026* (Nov. 1995). arXiv: quant-ph/9511026.

[Kit03] A. Yu. Kitaev. "Fault-Tolerant Quantum Computation by Anyons". In: *Annals of Physics* 303.1 (Jan. 2003), pp. 2–30.

[LMR13] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. "Quantum Algorithms for Supervised and Unsupervised Machine Learning". In: *arXiv:1307.0411 [quant-ph]* (Nov. 2013). arXiv: 1307.0411 [quant-ph].

[LMR14] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. "Quantum Principal Component Analysis". In: *arXiv* (July 2014).

[Man80] Yuri Manin. "Computable and Uncomputable". In: *Sovetskoye Radio, Moscow* 128 (1980).

[McC+16] Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. "The Theory of Variational Hybrid Quantum-Classical Algorithms". In: *New Journal of Physics* 18.2 (Feb. 2016), p. 023023.

[McC+18] Jarrod R. McClean, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven. "Barren Plateaus in Quantum Neural Network Training Landscapes". In: *Nature Communications* 9.1 (Dec. 2018), p. 4812. arXiv: 1803.11173.

[Ped+19] Edwin Pednault, John A. Gunnels, Giacomo Nannicini, Lior Horesh, and Robert Wisnieff. "Leveraging Secondary Storage to Simulate Deep 54-Qubit Sycamore Circuits". In: *arXiv:1910.09534 [quant-ph]* (Oct. 2019). arXiv: 1910.09534 [quant-ph].

[Per+14] Alberto Peruzzo et al. "A Variational Eigenvalue Solver on a Photonic Quantum Processor". In: *Nature Communications* 5.1 (July 2014), p. 4213.

[Pes+21] Arthur Pesah et al. "Absence of Barren Plateaus in Quantum Convolutional Neural Networks". In: *Physical Review X* 11.4 (Oct. 2021), p. 041011.

[Pre18] John Preskill. "Quantum Computing in the NISQ Era and Beyond". In: *Quantum* 2 (Aug. 2018), p. 79.

[RML14] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. "Quantum Support Vector Machine for Big Data Classification". In: *Physical Review Letters* 113.13 (Sept. 2014), p. 130503.

[SB09] Dan Shepherd and Michael J. Bremner. "Temporally Unstructured Quantum Computation". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 465.2105 (May 2009), pp. 1413–1439.

[Sho96] Peter W Shor. “Fault-Tolerant Quantum Computation”. In: *Proceedings of 37th Conference on Foundations of Computer Science*. IEEE. 1996, pp. 56–65.

[Sho94] P.W. Shor. “Algorithms for Quantum Computation: Discrete Logarithms and Factoring”. In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. Santa Fe, NM, USA: IEEE Comput. Soc. Press, 1994, pp. 124–134.

[Sim94] D. Simon. “On the Power of Quantum Computation”. In: *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. Los Alamitos, CA, USA: IEEE Computer Society, Nov. 1994, pp. 116–123.

[WBL12] Nathan Wiebe, Daniel Braun, and Seth Lloyd. “Quantum Algorithm for Data Fitting”. In: *Physical Review Letters* 109.5 (Aug. 2012), p. 050505.

[Zho+20] Han-Sen Zhong et al. “Quantum Computational Advantage Using Photons”. In: *Science* 370.6523 (Dec. 2020), pp. 1460–1463. arXiv: 2012.01625.