Applications of Quantum Computing in Algorithmic Time Complexity and Machine Learning

Research Report

ANIL VERCRUYSSE
ID 22202474

COMP47340: Computational Thinking

Word Count: 2 124

UCD School of Computer Science

20 November 2022

TABLE OF CONTENTS

1. Introduction

This research report was compiled for the Computational Thinking Module. As required, the report will explore two topics presented in the module and will do so by following a structured approach. The first topic that will be explored relates to Algorithmic Time Complexity, as it was presented in the Week 6 Lecture entitled ‘What is Big-O and why is it important?’. The second chosen topic relates to Machine Learning, as it was presented in the Week 10 Lecture entitled ‘Poems that solve puzzles’.

Both topics will first be introduced based on the course materials and additional academic sources. The report will then further explore the topics by examining some of the applications of Quantum Computing in both of these fields. It is important to note that the scope of this research report will be limited due to the constraints of the assignment. Specifically, the report will focus on the time complexity limitations of algorithms and will not explore other complexity factors. With regards to Machine Learning, only a limited number of Machine Learning models and techniques will be investigated. Lastly, given the high complexity of the theoretical framework underlying Quantum Computing, the last section of the report will aim to provide the reader with a concise introduction to this field and its applicability to the chosen topics.

2. Algorithmic Time Complexity

Computers have the ability to perform complex systematic computations and we rely on them to solve computing problems. While this reliance has been highly beneficial to mankind, it is important to understand the limitations that computers face when attempting to solve problems of exponentially increasing size. Computers are designed to solve problems through algorithms, which can be defined as a finite set of operations or methods for solving particular classes of information problems.[1] Typically, an algorithm will take a number of factors as input and will produce outputs based on its inherent operations. An algorithm’s complexity is typically measured based on factors such as its running time, its storage requirements, or any other unit of expense that it consumes.[2]

When selecting the appropriate algorithm to solve a problem, the most important factor consists in the running time of that algorithm. The running time of an algorithm consists in the amount of computer time it takes to run as a function.[3] Certain classes of problems are so intensive in computing power that they require absurd running times to be solved, if this is at all possible. An example of such a problem is the Travelling Salesman Problem (TSP), which aims at finding the least costly route that a travelling salesman may take to visit exactly once

[1] Wilf, H. S. (2002). *Algorithms and complexity*. AK Peters/CRC Press. p.1 and Algorithm definition & meaning. (2022) from https://www.merriam-webster.com/dictionary/algorithm
[2] Wilf, H. S. (2002). *Algorithms and complexity*. AK Peters/CRC Press. p.1
[3] Tornede, A., Wever, M., Werner, S., Mohr, F., & Hüllermeier, E. (2020, September). Run2survive: a decision-theoretic approach to algorithm selection based on survival analysis. In *Asian Conference on Machine Learning*. p.739

each of a list of m cities (where the cost of travelling from city i to city j is c_{ij}) before returning to his home city.[4] While we will not examine this problem in detail, we can use it to illustrate the exponential character of algorithmic time complexity. The problem seems straightforward enough with a small number of cities to visit, but the combinatorial character of the TSP means that even a slight increase in the value of m will result in exponentially increasing computing resources. For instance, if we were to find the optimal route for $m = 16$, there would be 653,837,184,000 distinct routes to evaluate individually.[5]

The time complexity of an algorithm is typically assessed by considering the worst-case time complexity of that algorithm, where the upper bound of its running time is taken as its measure of performance.[6] While this approach can often give an overly pessimistic estimation of the algorithm's time complexity, it is easier to compute that the average-case time complexity and can serve as a starting point to optimise an algorithm. Research on the time complexity of algorithms focuses on the asymptotic behaviour of their complexity, meaning the way in which the behaviour of an algorithm changes as its inputs grow.[7]

Time complexity can be expressed through Big-O notation, or Landau's symbol, which is a symbolism used in complexity to describe the asymptotic behaviour of function, namely the rate of growth at which the complexity of an algorithm increases.[8] Figure I below shows some examples of Big-O notations, which are ordered to first show the functions that have the lesser rate of growth before showing functions that have exponential growth rates. Big-O notations are used to capture the worst-case time complexity of an algorithm and therefore allow for the classification of algorithms in terms of their time complexity.[9] When looking back at the TSP described above we can see based on the table that it is a factorial problem, which requires one of the most time intensive types of algorithms.

---

[4] Hoffman, K. L., Padberg, M., & Rinaldi, G. (2013). Travelling salesman problem. *Encyclopedia of operations research and management science*, *1*, pp.1573-1578.
[5] Ibid
[6] Aho, A. V., & Ullman, J. D. (1992). *Foundations of computer science*. Computer Science Press, Inc. p.93
[7] McCool, M., Reinders, J., & Robison, A. (2012). *Structured parallel programming: patterns for efficient computation*. Elsevier. pp.65-67
[8] MIT Lecture Notes (2022) from http://web.mit.edu/16.070/www/lecture/big_o.pdf p.1
[9] Mohr, A. (2014). Quantum computing in complexity theory and theory of computation. *Carbondale, IL*, *194*. p.2

Figure I - Big-O Types of Order Hierarchy[10]

Notation	Name
$O(1)$	Constant
$O(\log(n))$	Logarithmic
$O(\log(\log(n))$	Double logarithmic (iterative logarithmic)
$o(n)$	Sublinear
$O(n)$	Linear
$O(n\log(n))$	Loglinear, Linearithmic, Quasilinear or Supralinear
$O(n^2)$	Quadratic
$O(n^3)$	Cubic
$O(n^c)$	Polynomial (different class for each $c > 1$)
$O(c^n)$	Exponential (different class for each $c > 1$)
$O(n!)$	Factorial
$O(n^n)$	- (Yuck!)

3. Machine learning

In recent years, there has been a strong interest in the development of machines that are able to learn in the same way that humans do. This led to the development of Machine Learning (ML), which can be defined as 'the use and development of computer systems that are able to learn and adapt without following explicit instructions, by using algorithms and statistical models to analyse and draw inferences from patterns in data'.[11] ML is a branch of the field of Artificial Intelligence, which more generally refers to 'the general ability of computers to emulate human thought and perform tasks in real-world environments'.[12] ML is one of the most promising technologies of our times and its versatility allows it to play a vital role in many sectors. For instance, ML models have been used to forecast the spread of the COVID-19 pandemic, but they have also been applied to financial markets to optimise high frequency trading algorithms.[13]

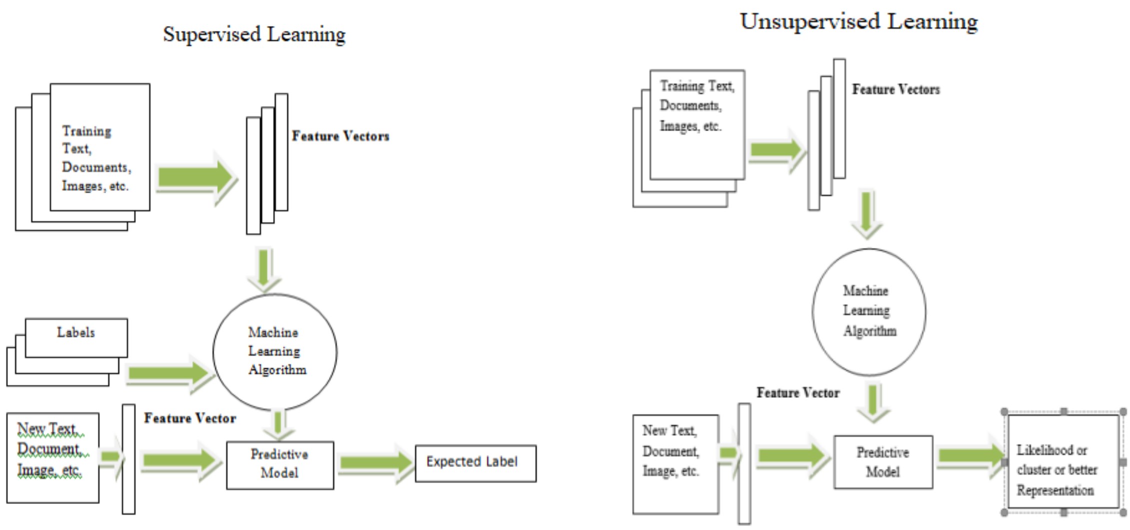
The starting point of ML is data, which must be prepared and aggregated to serve as input for a ML algorithm so that it may produce information in a specific output format.[14] In the process of formatting data to serve as input, a choice of the relevant data features should be made to remove features that are irrelevant to the algorithm's learning process. Once the data has been adequately processed, the ML algorithm that is best suited to process the data should

[10] Big-O notation (2010) from http://www.scs.ryerson.ca/~mth110/Handouts/PD/bigO.pdf
[11] IMB AI and Machine Learning Platform Integration (2022) from https://www.ibm.com/docs/en/cloud-paks/1.0?topic=cloudpaks_start/ibm-process-mining/user-manuals/ai_ml_platformintegration/introduction.htm
[12] Artificial Intelligence - Columbia Engineering (2022) https://ai.engineering.columbia.edu/ai-vs-machine-learning/#:~:text=Put%20in%20context%2C%20artificial%20intelligence,and%20improve%20themselves%20through%20experience
[13] Kearns, M., & Nevmyvaka, Y. (2013). Machine learning for market microstructure and high frequency trading. *High Frequency Trading: New Realities for Traders, Markets, and Regulators*. and Lalmuanawma, S., Hussain, J., & Chhakchhuak, L. (2020). *Applications of machine learning and artificial intelligence for Covid-19 (SARS-CoV-2) pandemic: A review*.
[14] Alzubi, J., Nayyar, A., & Kumar, A. (2018, November). Machine learning from theory to algorithms: an overview. In *Journal of physics: conference series* (Vol. 1142, No. 1, p. 012012). IOP Publishing. p.5

be chosen, along with its framework model and relevant parameters.[15] After these initial steps, the ML algorithm is trained on a part of the data set and an evaluation of its performance is undertaken to adjust its parameters before feeding the algorithm with the full data set.

There are three main types of ML models: supervised, unsupervised, and reinforced learning.[16] Supervised learning (SL) can be understood as 'learning by example' as it involves feeding the ML algorithm with example outputs in relation to input data, with the expectation that the algorithm will be able to recognise the pattern through which correct outputs were obtained.[17] As opposed to this, unsupervised learning (UL) models will not provide a ML algorithm with any target outputs but will instead feed it unlabeled data with the goal of finding unidentified patterns through statistical and probabilistic models.[18] Reinforced learning serves as a sort of intermediate model between SL and UL in that it is similar to UL, but has the addition of providing the ML algorithm with feedback in relation to the output it produces.[19]

Figure II - Supervised and Unsupervised Learning Diagrams[20]



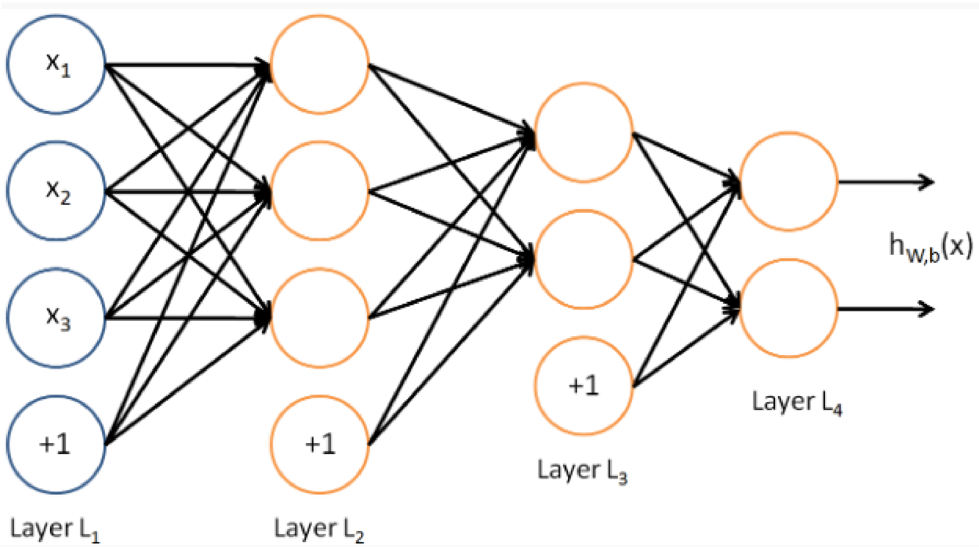
Within the realm of ML techniques, the subfield of Deep Learning (DL) has become a focus of recent ML research. DL can be defined as 'an advanced type of machine learning that uses multilayered neural networks to establish nested hierarchical models for data processing and analysis, with the goal of self-directed information processing'.[21] DL makes use of Artificial

---

[15] Ibid
[16] Ayodele, T. O. (2010). Types of machine learning algorithms. *New advances in machine learning*, *3*, pp.19-48.
[17] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *Overview of supervised learning. In The elements of statistical learning* (pp. 9-41). Springer, New York, NY. p.15
[18] Ghahramani, Z. (2003, February). *Unsupervised learning. In Summer school on machine learning* (pp. 72-112). Springer, Berlin, Heidelberg. pp.3-4
[19] Alzubi, J., Nayyar, A., & Kumar, A. (2018, November). *Machine learning from theory to algorithms: an overview*. In Journal of physics: conference series (Vol. 1142, No. 1, p. 012012). IOP Publishing. p.7
[20] Ibid
[21] Deep Learning Definition (2022) from https://www.dictionary.com/browse/deep-learning

Neural Networks (ANN), which simulate a biological network of neurons through a computer.[22] ANN units will receive inputs from other units until a defined threshold is met, which will eventually lead the initial unit to emit its own output to other units.[23] ANN have multiple layers, which allows them to contain many parameters and to model complex functions.[24] ANN allow for the recognition of complex non-linear relationships and are able to generalise, meaning that they can apply their learnings to data sets that are distinct from those they were trained with.[25]

Figure III - Multi-Layer Neural Network[26]



4. Applications of Quantum Computing in Algorithmic Time Complexity and Machine Learning

a. Quantum Computing and Algorithmic Time Complexity

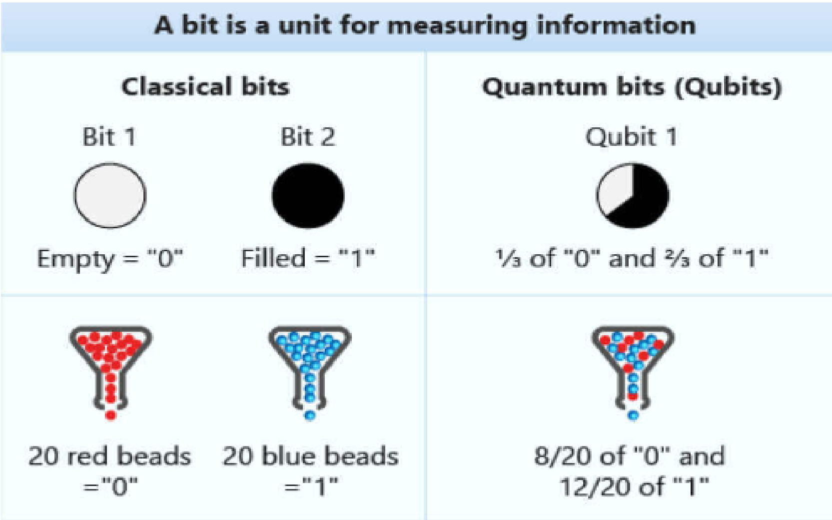
As we have seen in Section 2 of this report, classical computers encounter difficulties when tasked with solving certain classes of complex problems.[27] This limitation is in part due to the Boolean model of classical computers, which limits the value of each basic information unit, or bit, to be either 0 or 1.[28] In such a model, problems that have a high number of input

[22] Krogh, A. (2008). *What are artificial neural networks?*. Nature biotechnology, 26(2), pp.195-197.
[23] Ibid
[24] Deep Learning Stanford - Multi-Layer Neural Network (2022) from http://deeplearning.stanford.edu/tutorial/supervised/MultiLayerNeuralNetworks/
[25] Liu, Z., Lian, T., Farrell, J., & Wandell, B. A. (2020). *Neural network generalization: The impact of camera parameters.* IEEE Access, 8, pp.10443-10454.
[26] Deep Learning Stanford - Multi-Layer Neural Network (2022) from http://deeplearning.stanford.edu/tutorial/supervised/MultiLayerNeuralNetworks/
[27] What is Quantum Computing - Microsoft Azure (2021) from https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-quantum-computing/#introduction
[28] Solving Classical Computing Problems Via Quantum Computing University of South Florida (2021) https://unfsoars.domains.unf.edu/2021/posters/solving-classical-computing-problems-via-quantum-computing/

factors will quickly overwhelm the input capacity of a classical computer and require absurd computation times. In this context, the emerging field of quantum computing offers promising avenues to massively increase computation speeds.

Instead of relying on Boolean values, quantum computers are based on principles of quantum mechanics, which allows them to exploit the properties of subatomic particles. The basic information units of quantum computers are called qubits. Qubits are able to be superposed into multiple states, which means that they can store multiple values at the same time.[29] This superposition property can be exploited by quantum algorithms, which are algorithms designed for quantum computers, to attain computing powers that are exponentially superior to classical computers.[30]

Figure IV - Bits vs Qubits[31]



While the technology that underlies quantum computers has not yet been fully developed, it could provide significant improvements for resolving problems with high time complexity. It is however important to note that quantum computers will not be able to solve all classes of problems. This is due to the fact that a quantum computer can only run probabilistic algorithms, because of the non-deterministic nature of quantum mechanics.[32] The non-deterministic nature of quantum mechanics entails that it is not possible to obtain precise knowledge at the quantum level.[33] Therefore, only classes of problems that are compatible with probabilistic algorithms can be solved with quantum computers, which in turn means that they are not suited to resolve all instances of algorithmic time complexity.

[29] Steane, A. (1998). *Quantum computing.* Reports on Progress in Physics, 61(2), 117. p.25
[30] What is a Qubit - Microsoft Azure (2021) from https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-a-qubit/#qubit-vs-bit
[31] Ibid
[32] Mohr, A. (2014). *Quantum computing in complexity theory and theory of computation*. Carbondale, IL, 194. p.3
[33] Bergstra, J. A., Ponse, A., & Smolka, S. A. (Eds.). (2001). *Handbook of process algebra*. Elsevier.

b. Quantum Machine Learning

Quantum computing also offers possibilities in the field of ML, whereby quantum algorithms are developed to solve ML problems with the efficiency of quantum computers.[34] Quantum computing vastly improves computing speeds, which can be useful in ML given that the training of complex models tends to entail high computational costs.[35] The synergy between these fields has led to their integration through Quantum machine Learning (QML) which can be defined as 'an area of research that focuses on the translation of machine learning algorithms into quantum circuits that use qubits instead of binary bits'.[36] Quantum circuits are operations that can change the superposition state of qubits.[37]

An example of the potential of quantum computers in the ML field can be made when examining a nonlinear problem. A nonlinear problem refers to a situation where there isn't a direct relation between data elements, which makes it harder to classify this data into distinct groups.[38] Typically, nonlinear data sets require the use of a kernel function, which is a ML technique that separates such data elements into distinct groups.[39] A problem encountered when using kernel functions is the high runtime they require when dealing with complex data sets.[40] Quantum computers can aid this process by producing quantum kernels functions, which have been proven to reduce runtime and improve performance for certain classes of problems.[41]

5. Conclusion

The goal of this report was to overview the topics of Algorithmic Time Complexity and ML, and to explore the potential applications of Quantum Computing in these fields.

We have seen that the selection of an algorithm is based on its worst-case time complexity scenario, and that this determination can be aided by means of Big-O classification. Quantum Computing was identified as a potential solution to high computation times, but it was shown that this technology is not perfectly suited for all problem classes.

---

[34] Schuld, M., Sinayskiy, I., & Petruccione, F. (2015). *An introduction to quantum machine learning.* Contemporary Physics, 56(2), pp.172-185.
[35] What is Quantum Computing - Microsoft Azure (2021) from https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-quantum-computing/#introduction
[36] Quantum ML (2021) from https://www.techopedia.com/definition/34634/quantum-ml
[37] P.A.M. Dirac (1947). *The Principles of Quantum Mechanics* (2nd ed.). Clarendon Press. p.12.
[38] Quantum Machine Learning Explained - IBM Youtube from https://www.youtube.com/watch?v=NqHKr9CGWJ0 and Explained: Linear and nonlinear systems - MIT News from https://news.mit.edu/2010/explained-linear-0226
[39] Hofmann, T., Schölkopf, B., & Smola, A. J. (2008). *Kernel methods in machine learning.* The annals of statistics, 36(3), p.1180.
[40] Ibid
[41] Quantum kernels can solve machine learning problems that are hard for all classical methods - IBM (2021) from https://research.ibm.com/blog/quantum-kernels

We also observed that ML can be a powerful tool to identify patterns in large sets of data and that various ML models can be used for different purposes. More advanced forms of ML, such as DL, even go a step further in this process by attempting to emulate biological neurons to recognise complex relationships and generalise their learnings. Synergies between Quantum Computing and ML were also identified, as the former can produce quantum kernel functions that in particular cases improves the training time and performance of ML algorithms.

Overall, it was interesting to note that the fields of Algorithmic Time Complexity, ML, and Quantum Computing present many interactions, in particular as time complexity tends to be a key performance indicator in computing systems.

BIBLIOGRAPHY

Academic Journals

Wilf, H. S. (2002). *Algorithms and complexity*. AK Peters/CRC Press.

Tornede, A., Wever, M., Werner, S., Mohr, F., & Hüllermeier, E. (2020, September). Run2survive: a decision-theoretic approach to algorithm selection based on survival analysis. In *Asian Conference on Machine Learning*.

Hoffman, K. L., Padberg, M., & Rinaldi, G. (2013). Travelling salesman problem. *Encyclopedia of operations research and management science*, *1*.

Aho, A. V., & Ullman, J. D. (1992). *Foundations of computer science*. Computer Science Press, Inc.

McCool, M., Reinders, J., & Robison, A. (2012). *Structured parallel programming: patterns for efficient computation*. Elsevier.

Mohr, A. (2014). Quantum computing in complexity theory and theory of computation. *Carbondale, IL*, *194*.

Kearns, M., & Nevmyvaka, Y. (2013). Machine learning for market microstructure and high frequency trading. *High Frequency Trading: New Realities for Traders, Markets, and Regulators.*

Lalmuanawma, S., Hussain, J., & Chhakchhuak, L. (2020). *Applications of machine learning and artificial intelligence for Covid-19 (SARS-CoV-2) pandemic: A review*.

Alzubi, J., Nayyar, A., & Kumar, A. (2018, November). Machine learning from theory to algorithms: an overview. In *Journal of physics: conference series* (Vol. 1142, No. 1, p. 012012). IOP Publishing.

Ayodele, T. O. (2010). Types of machine learning algorithms. *New advances in machine learning*, *3*.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *Overview of supervised learning. In The elements of statistical learning*. Springer, New York, NY.

Ghahramani, Z. (2003, February). *Unsupervised learning. In Summer school on machine learning* (pp. 72-112). Springer, Berlin, Heidelberg.

Krogh, A. (2008). *What are artificial neural networks?*. Nature biotechnology, 26(2).

Liu, Z., Lian, T., Farrell, J., & Wandell, B. A. (2020). *Neural network generalization: The impact of camera parameters.* IEEE Access, 8.

Steane, A. (1998). *Quantum computing.* Reports on Progress in Physics, 61(2), 117.

Mohr, A. (2014). *Quantum computing in complexity theory and theory of computation*. Carbondale, IL, 194.

Schuld, M., Sinayskiy, I., & Petruccione, F. (2015). *An introduction to quantum machine learning.* Contemporary Physics, 56(2).

Hofmann, T., Schölkopf, B., & Smola, A. J. (2008). *Kernel methods in machine learning.* The annals of statistics, 36(3).

Course Books

Bergstra, J. A., Ponse, A., & Smolka, S. A. (Eds.). (2001). *Handbook of process algebra*. Elsevier.

P.A.M. Dirac (1947). *The Principles of Quantum Mechanics* (2nd ed.). Clarendon Press.

Lecture notes

MIT Lecture Notes (2022) from http://web.mit.edu/16.070/www/lecture/big_o.pdf p.1

Big-O notation (2010) from http://www.scs.ryerson.ca/~mth110/Handouts/PD/bigO.pdf

Deep Learning Stanford - Multi-Layer Neural Network (2022) from http://deeplearning.stanford.edu/tutorial/supervised/MultiLayerNeuralNetworks/

Webpages

Algorithm definition & meaning. (2022) from https://www.merriam-webster.com/dictionary/algorithm

IMB AI and Machine Learning Platform Integration (2022) from https://www.ibm.com/docs/en/cloud-paks/1.0?topic=cloudpaks_start/ibm-process-mining/user-manuals/ai_ml_platformintegration/introduction.htm

Artificial Intelligence - Columbia Engineering (2022) https://ai.engineering.columbia.edu/ai-vs-machine-learning/#:~:text=Put%20in%20context%2C%20artificial%20intelligence,and%20improve%20themselves%20through%20experience

Deep Learning Definition (2022) from https://www.dictionary.com/browse/deep-learning

What is Quantum Computing - Microsoft Azure (2021) from https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-quantum-computing/#introduction

Solving Classical Computing Problems Via Quantum Computing University of South Florida (2021) https://unfsoars.domains.unf.edu/2021/posters/solving-classical-computing-problems-via-quantum-computing/

What is a Qubit - Microsoft Azure (2021) from https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-a-qubit/#qubit-vs-bit

Quantum ML (2021) from https://www.techopedia.com/definition/34634/quantum-ml

Quantum Machine Learning Explained - IBM (2022) Youtube from https://www.youtube.com/watch?v=NqHKr9CGWJ0

Explained: Linear and nonlinear systems - MIT News (2010) from https://news.mit.edu/2010/explained-linear-0226

Quantum kernels can solve machine learning problems that are hard for all classical methods - IBM (2021) from https://research.ibm.com/blog/quantum-kernels