2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)
Amity University, Noida, India. Oct 13-14, 2022

2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO) | 978-1-6654-7433-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICRITO56286.2022.9964828

An Implementation of Quantum Machine Learning Technique to Determine Insurance Claim Fraud

Subodh Nath Pushpak[#*]
[#]*Amity Institute of Information Technology (AIIT),*
Amity University
Noida, Uttar Pradesh, India
subodh.pushpak@s.amity.edu

Sarika Jain[*]
Amity Institute of Information Technology(AIIT),
Amity University
Noida, Uttar Pradesh, India
sjain@amity.edu

***Abstract*—Quantum Machine Learning (QML) is an upcoming and active research field that uses quantum computing concepts for machine learning. This paper presents QML techniques for detecting fraud in housing insurance claims. Insurance fraud detection is challenging as the patterns may be complex. Therefore, various techniques and methods are employed to identify suspicious claims and prevent business losses due to these fraudulent claims. This paper explores Quantum Support Vector Machine (QSVM) techniques and feature engineering, feature selection, and parameter tweaking to identify fraudulent housing insurance claims. In addition, it compares Quantum SVM with classical SVM. Finally, this paper focuses on detecting property insurance claims fraud by using machine learning techniques.**

Keywords: Quantum machine learning, Fraud Detection, Quantum Support Vector Machine (QSVM), Quantum computing.

I. Introduction

Quantum machine learning is a new approach to machine learning based on the quantum physics principle of superposition and entanglement. Quantum machine learning lends itself to a wide variety of techniques implemented for machine learning. The quantum phenomenon of entanglement and superposition makes quantum computing a desirable candidate for machine learning algorithms [1]. For example, QML is used in machine learning tasks like image classification, feature finding, quantum support vector machines, quantum clustering techniques, and more. In addition, quantum computing is particularly advantageous in cases where the flow of machine learning algorithms can be parallelized. We have used IBM® Quantum Experience to interact with actual quantum computers over the cloud.

A. *Machine Learning*

Machine learning is a technique of training computers to predict results while learning from the data without being programmed explicitly. This allows them to predict outcomes rather than just follow a set of instructions. Machine learning algorithms are used to find complex patterns and analyze vast amounts of data. These patterns can then be used to predict future outcomes. There are three kinds of machine learning methods:

Supervised learning. It is a technique that works on data that is already labeled with some of its features.

Unsupervised learning is used to analyze the data based on similarities and dissimilarities of their classes. No labeled data is present.

Reinforcement learning- It is a technique that uses reward and penalty feedback to learn.

B. *Quantum Computing*

Quantum computing makes use of quantum mechanical phenomena to perform computations on data. The classical computer works on data is represented as bits, i.e., a string of 1s and 0s, similarily to quantum computing; information is represented by quantum bits or qubits. A qubit represents 1, 0, or any quantum superposition of these two qubits. Due to qubit representation, n qubits can be in 2^n different states simultaneously. 2^N bits of classical information can be obtained from the N-qubit quantum system. Two properties of a quantum system that stands apart from classical are:

Superposition: It signifies that a quantum system may be in two different states simultaneously. However, it's not until it is measured that it will be in one specific condition.

Entanglement: It is a quantum phenomenon that occurs when two or more objects are inextricably linked together, even if large distances separate them. This linkage leads to correlations between the observable physical properties of the systems.

While quantum computing does not provide speedup for all types of problems, it has been mathematically proven to provide efficient solutions to certain kinds of problems [2]. Quantum computers effectively solve complex problems, especially in optimization and simulation[5], [6]. They can approximate the properties of partition functions, perform approximate optimization, and simulate different quantum systems.

Quantum computing is different than classical computing and has exotic properties. Below is a brief reference of its concepts which is helpful for this paper.

The classical computing state can assume any one of the possible states of $\{0,1\}$, whereas, in quantum computing, the fundamental state is represented via quantum bits (qubits). A Qubit is a unit (normalized) vector in a two-dimensional complex vector space [3]. Mathematically, a qubit is a vector such that

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix}\alpha \\ \beta\end{bmatrix}$$

such that $\alpha,\ \beta \in \mathbb{C}$,

$$|\alpha|^2 + |\beta|^2 = 1$$

ℂ represents the set of complex numbers. Two special quantum states corresponding to the 0 and 1 states of a classical bit are represented as ket vectors

$$|0\rangle := \begin{bmatrix}1\\0\end{bmatrix} \qquad |1\rangle := \begin{bmatrix}0\\1\end{bmatrix}$$

A qubit may be in superposition with itself, or two or more qubits can be in entanglement. The qubit state is manipulated through unitary operations of the quantum gates.

TABLE I. QUANTUM GATES AND THEIR MATRIX REPRESENTATION

Sno	Name of Gate	Matrix representation
1	X Gate or Pauli-X	$\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$
2	Y Gate or Pauli -Y	$\begin{bmatrix}0 & -i\\i & 0\end{bmatrix}$
3	Z Gate or Pauli - Z	$\begin{bmatrix}1 & 0\\0 & -1\end{bmatrix}$
4	H Gate or Hadamard	$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}$
5	S Gate or Phase	$\begin{bmatrix}1 & 0\\0 & i\end{bmatrix}$
6	T Gate / (π/8)	$\begin{bmatrix}1 & 0\\0 & e^{i\pi/4}\end{bmatrix}$
7	CX Gate / Controlled Not	$\begin{bmatrix}1 & 0 & 0 & 0\\0 & 1 & 0 & 0\\0 & 0 & 0 & 1\\0 & 0 & 1 & 0\end{bmatrix}$
8	CZ Gate / Controlled Phase	$\begin{bmatrix}1 & 0 & 0 & 0\\0 & 1 & 0 & 0\\0 & 0 & 1 & 0\\0 & 0 & 0 & -1\end{bmatrix}$

A. Quantum Machine Learning

Quantum Machine Learning (QML) studies how quantum computing can be used to improve machine learning algorithms. In particular, quantum machine learning focuses on how quantum computers can be used to enhance the speed and accuracy of machine learning algorithms[7], [8]. One of the primary advantages of quantum machines is that they can exploit the exponential number of states that can be represented in a quantum system. This allows for more accurate modeling of complex data sets and improves performance in machine learning tasks. Additionally, quantum machines can be used to better approximations of difficult-to-compute kernel functions, enhancing the speed and accuracy of many machine learning algorithms. Quantum machine learning is still an emerging field and many open questions about the best ways to use quantum computers for machine learning tasks. However, the potential for quantum machines to enhance the performance of machine learning algorithms is clear, and research in this area is rapidly expanding.

II. FRAUD DETECTION

Fraud is the intent to deceive or obtain an unearned benefit. The estimated total cost of insurance fraud is estimated to be more than forty billion dollars. In addition, substantial financial losses occur in various industries across insurance, banking, transportation, communications, manufacturing, healthcare, and retail sectors each year due to Fraudulent activities.

A. Fraud

Many new technologies have been introduced as antifraud mechanisms detecting nonobvious relationships and associations. Effective data analysis leads to lower fraud incidence and better pattern detection.

Patterns are repetitive combinations, sequences, or relationships of values that have significance within the context or domain for which they are being reviewed. There are many ways to detect patterns that can be very complex for analysis. For example, the detection of fraud is a challenging problem. Fraud detection is based on analyzing and developing heuristics around fraud patterns and indicators.

Deep and fundamental expertise is often required to discover patterns of data that are of interest. The application of fraud analytics requires domain knowledge and understanding of the correlation of the data, along with the skills and abilities to identify the recognized pattern or anomaly in the data. Fraud detection and prevention are related. Prevention means stopping fraud, whereas Fraud detection means sensing whether or not fraud has happened The algorithms and tools analyze transactional data and data relationships for signs of fraudulent activity or some anomaly to identify fraud.

One of the ways to avoid fraud is to reduce the surface area of the possible fraud exposures. Early detection and prompt action can reduce the losses due to fraud. Detection of fraud involves determining the symptoms and identifying the exposures. Fraud detection can be tricky since the presence of patterns that indicates fraud may not mean always mean that fraud exists, and the absence of symptoms does not imply that there is no fraud. It is very crucial to determine the cause and symptoms of the correlations of fraud and prevent them.

The most crucial step in detecting fraud is the analysis of the data to identify if the signs of fraud are present. These signs can be identified with the aid of machine learning techniques.

III. QUANTUM MACHINE LEARNING TECHNIQUES

When the feature space becomes large, it can be difficult to find a good solution using traditional methods. Quantum algorithms can be much faster because they take advantage of the large number of possible states that can be explored using quantum mechanics. Quantum state-space means a set of all possible combinations of qubits. The feature space can be expressed via quantum state space and can be used to express a classifier in supervised learning[11].

The data needs to be transformed before it can be used in quantum machine learning algorithms. The process of mapping classical data to a quantum computer is known as the state preparation problem. The choice of the type of state preparation has a direct impact on the choice of machine learning algorithms in terms of operations utilized in the quantum algorithm and the computational complexity of the entire machine learning algorithm.

A. State preparation for Quantum Processing

While comparing classical and quantum machine learning algorithms, the algorithmic complexity of state preparation should be considered. The state preparation stage is of the order of the degree of freedom of the data. Primarily, there are three types of encoding. The Basic, Amplitude, and the Hamiltonian encoding.

B. The quantum Circuit

A quantum algorithm is executed in the form of a circuit that performs operations on qubits via quantum gates. A typical quantum circuit consists of three stages.

1) *Initialize the circuit by setting the qubits to the |0> stage.*
2) *Perform operations on qubits by applying quantum gates to the qubits*
3) *Read out the information from qubits by measurement typically to a classical bit.*

Each quantum algorithm is represented as a circuit that can be executed on a quantum computer or simulator. We now consider a machine learning algorithm that can be employed to detect fraud by classifying data.

IV. Quantum Support Vector machines

Support vector machine algorithms can be adapted to quantum counterparts. In fact, both classical and quantum may also be used simultaneously, where quantum may provide for kernel of the SVM algorithm.

A. Background and Motivation

The support vector machine (SVM) is a supervised learning algorithm used for data classification and pattern recognition [4]. SVM is trained using the data and is used to forecast whether the input belongs to a special.SVM works by creating a model that maps input data to a higher-dimensional space, where the patterns in the data are easier to identify. Once the model is created, it can be used to predict values for new data points to classify or predict objects in a given dataset. In order to do so, the SVM requires two input parameters: (1) a set of training data and (2) a classification function. The training data is a collection of objects (e.g., images, text documents, etc.) along with their associated class labels [5]. The classification function is a mathematical function that takes as input an object and outputs a single numerical value indicating the class to which that object belongs. The SVM finds the decision boundary that maximizes the distance between points in different groups. The decision boundary is a mathematical function that separates the training data into distinct groups. The kernel function then calculates the distance between every point in the training dataset and every point on the decision boundary. The support vectors are data points used to maximize the distance between classes. In essence; the SVM algorithm evaluates the values of parameters w and b such that :

$$wx + 1 > 1; \text{ for data with labels } y = 1$$
$$wx + 1 < 1; \text{ for data with labels } y = -1$$

The generalization of the above is analogous to the optimization of a quadratic problem with linear constraints.

$$min_{w,} \; ||w||^2$$

$$\text{so that } \; y_i \, (w_i + b) >= i$$

$$\text{where } i = 1..\, N$$

The optimization problem of the SVM is equivalent to the minimization of its dual formulation. The Local maxima and minima of a function are evaluated using Lagrange multipliers subject to equality constraints.

The dual form is helpful because dot products can be replaced by a kernel function that implicitly encodes the feature map[13]. This technique is called the kernel trick, and it is used in support vector machines to reduce the computational complexity of SVMs. In addition, non-linearly separable datasets may become linearly separable by including new features.

B. Quantum Support Vector Machine

The quantum support vector machine is an extension of the classical support vector machine. It uses quantum techniques while implementing SVM. QSVM algorithm can be implemented using several approaches on quantum machines. Grover's search algorithm can be used in SVM to provide quadratic speedup. The quantum variational classifier uses a variational quantum eigensolver (VQE) circuit to classify the data as the classic SVM.

V. Implementation of QSvM For Fraud detection

Generating a QSVM-based algorithm requires the following steps.

1) *Scale and normalize the data.*
2) *Apply principal component analysis to reduce dimensions. This is a crucial step as the number of qubits available may be limited.*
3) *Generate a kernel.*
4) *Estimate the kernel for a set of test data.*
5) *Simulate and execute the quantum circuit on quantum computers available over the cloud.*
6) *Predicting the outcome for new data points*

The choice of which feature map to use is important and may depend on the given dataset we want to classify

VI. Quantum Feature maps

The quantum feature map can be represented as $\phi(x)$ from the classical feature vector x to some quantum state $|\Phi(x)\rangle\langle\Phi(x)|$. This quantum state is achieved by applying the unitary operation, UΦ(x) , on the initial state |0⟩n of the qubits, where n is the number of qubits used for encoding.

The feature maps are:

1) Pauli Feature Map

The Pauli Expansion circuit describes the unitary operator of depth d, which contains layers of Hadamard gates with entangled blocks. It has repetitions of the below transformation

$$U_{\Phi(\mathbf{x})} = \exp\left[i \sum_{S \subseteq [n]} \phi_S(\mathbf{x}) \prod_{k \in S} P_i \right]$$

Where Pi ∈ {I, X, Y, Z} are the Pauli matrices, S index is he measure of connectivity between qubits/data points represented by the data mapping ϕ_S

$$\phi_S(\mathbf{x}^{\mapsto}) = \begin{cases} x_i & \text{if } S = \{i\} \\ (\pi - x_i)(\pi - x_j) & \text{if } S = \{i, j\} \end{cases}$$

2) Z Feature Map

This is a sub-class of the Pauli Feature Map where the Pauli strings are fixed as Z matrices. The first-order expansion is a circuit without entangling gates. So when k=1 and $P_0 = Z$

$$\mathcal{U}_{\Phi(\mathbf{x})} = \left(\exp\left(i \sum_j \phi_{\{j\}}(\mathbf{x}) \, Z_j \right) H^{\otimes n} \right)^d$$

3) ZZ Feature Map

ZZFeatureMap is hard to replicate classically and is implemented as short-depth circuits on near-term quantum devices. In case k = 2, $P_0 = Z$ and $P_1 = ZZ$.

$$\mathcal{U}_{\Phi(\mathbf{x})} = \left(\exp\left(i \sum_{jk} \phi_{\{j,k\}}(\mathbf{x}) \, Z_j \otimes Z_k \right) \exp\left(i \sum_j \phi_{\{j\}}(\mathbf{x}) \, Z_j \right) H^{\otimes n} \right)^d$$

VII. QUANTUM KERNEL ESTIMATION

A quantum feature map represented b ϕ(x), is a quantum kernel,

$$k(x_i,x_j)=\phi(x_j)^{\dagger}\phi(x_i)$$

The quantum kernel can be represented as a matrix. Each element of this kernel matrix on a quantum computer may be evaluated by calculating the transition amplitude. The feature map is represented as a parameterized quantum circuit, that is, a unitary transformation (**x**) on n qubits.

Thus, the quantum kernel matrix is used in a classic kernel of a machine learning algorithm, such as support vector classification. This paper used this method to determine fraud using quantum machine learning.

VIII. FRAUD DETECTION USING QSVM

Below data parameters were used in the data to detect insurance fraud

TABLE II. EXTRACTED SIGNIFICANT FEATURES FROM DATA FOR FRAUD DETECTION

Sno	Field Name
1	Credit Score
2	Payment history
3	Outstanding debts
4	Property Age
5	Max Days Property Unoccupied
6	Number of Claims Last 3 years
7	Number of Claims in Current term
8	The gap between Loss & Intimation
9	The gap from the Effective date
10	Gap before the Expiration date
11	Authorities contacted
12	Police/Fire/Medical report available
13	Witnesses

IX. CONCLUSIONS

State preparation for the QML algorithms can be challenging. Below is the comparative accuracy of classical SVM and QSVM on the same dataset for detecting fraud in the insurance domain.

TABLE III. RESULTS FOR FRAUD DETECTION USING SUPPORT VECTOR MACHINES

Sno	SVM Name	Kernel / Entanglement	Accuracy (%)
1	Classic SVM	linear	98.57
2	Classic SVM	rbf	98.13
3	Classic SVM	poly	99.18
4	Classic SVM	sigmoid	98.44
5	QSVM (ZZfeatureMap)	linear	91.15
6	QSVM(ZZfeatureMap)	circular	92.66
7	QSVM(ZZfeatureMap)	full	92.67

The results and the observation in experimentation suggest that QSVM could potentially provide computation speedup. Though it is yet to be mathematically proven that quantum techniques provide a quantum advantage, the advancement in quantum hardware and the discovery of new approaches will provide a very good alternative to classical machine learning techniques.

REFERENCES

[1] H. J. Briegel and V. Dunjko, "Machine learning & artificial intelligence in the quantum domain: a review of recent progress," IOP Publishing Ltd, 2018.

[2] S. Kak, "Quantum Neural Computing," Advances in Imaging and Electron Physics,259-313, vol. 94, no. 1995.

[3] V. Vedral, T. Haug, L.-C. Kwek and K. Bharti, "Machine Learning meets Quantum Foundations: A Brief Survey," AVS Quantum Sci. 2, Vols. 034101, 2, no. 2639-0213, p. 17, 2020.

[4] R. Singh, N. Rajpal and R. Mehta, "Application-Specific Discriminant Analysis of Cardiac Anomalies Using Shift-Invariant Wavelet Transform," International Journal of E-Health and Medical Communications (IJEHMC), no. 10.4018/IJEHMC.20210701.oa5, p. 21, 2021.

[5] R. Singh, N. Rajpal and R. Mehta, "An Empiric Analysis of Wavelet-Based Feature Extraction on Deep Learning and Machine Learning Algorithms for Arrhythmia Classification," International Journal of Interactive Multimedia and Artificial Intelligence, vol. 6, no. 1989-1660 10.9781/ijimai.2020.11.005, 2021.

[6] N. Wiebe, A. Kapoor and K. Svore, “Quantum Algorithms for Nearest-Neighbor Methods for Supervised and Unsupervised Learning,” Quantum Information & Computation , vol. 15, p. 31, 2014.

[7] N. Wiebe, A. Kapoor and K. Svore, “Quantum Algorithms for Nearest-Neighbor Methods for Supervised and Unsupervised Learning,” eprint arXiv:1401.2142, vol. 15, p. 31, 2014.

[8] J. M. T. H. J. B. Vedran Dunjko, “Quantum-Enhanced Machine Learning,” Phys. Rev. Lett., vol. 117, no. 1079-7114, p. 13, 2016.

[9] M. S. Petruccione, Supervised Learning with Quantum Computers. Quantum Science and Technology, Springer, Cham, 2018.

[10] K. M. R. H. e. a. Mishra N, “Quantum Machine Learning: A Review and Current Status,” Advances in Intelligent Systems and Computing, vol. 1175, 2020.

[11] I. S. F. P. Maria Schuld, “Simulating a perceptron on a quantum computer,” Physics Letters A, vol. 379, no. 0375-9601, p. 660–663, 2015.

[12] I. S. F. P. M. Schuld, “An Introduction to Quantum Machine Learning Algorithms,” Contemporary Physics, 56:2, 172-185 (2015), vol. 56, no. 1366-5812, p. 19, 2015.

[13] S. Llyod, M. Mohseni and P. Rebentrost, “Quantum algorithms for supervised and unsupervised,” arXiv:1307.0411v2 [quant-ph], p. 11, 2013.

[14] P. Wittek, Quantum Machine Learning: What Quantum Computing Means to Data Mining, Academic Press, 2014.

[15] L. G. Valiant, “A theory of the learnable,” Commun. ACM 27(11), no. 1884, p. 1134–1142.

[16] M. Schuld, “Quantum machine learning for supervised pattern recognition,” University of KwaZulu-Natal, Durban 2017, 2018.

[17] R. de Wolf and S. Arunachalam, “Optimal Quantum Sample Complexity of Learning Algorithms,” Physical Review X, vol. 105, 2016.

[18] Mohseni, P. Rebentrost, S. Lloyd and Masoud, “Quantum Support Vector Machine for Big Data Classification,” Physical Review Letters, Vols. 1079-7114 113, 2014.

[19] J. Realpe-Gómez, A. Perdomo-Ortiz, R. Biswas and M. Benedetti, “Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers,” Quantum Science and Technology 3, vol. 3, no. 2058-9565, p. 30, 2018).

[20] P. by Xanadu, “Quantu ML,” Xanadu, 10 02 2020. [Online]. Available: https://pennylane.ai/qml/whatisqml.html.

[21] “Azure Quantum,” Microsoft, [Online]. Available: https://azure.microsoft.com/en-in/services/quantum/. [Accessed 10 02 2020].

[22] T. Li, “Quantum algorithms for machine learning and optimization,, Doctor of Philosophy,” Digital repository at the University of Maryland, 2020.

[23] T. Kadowaki and H. Nishimori, “Quantum annealing in the transverse ising model,” Vols. Phys. Rev. E 58, 5355–5363, 1998.

[24] IBM, “Quantum Experience,” [Online]. Available: https://quantum-computing.ibm.com/. [Accessed 10 02 2020].

[25] Google, “CIRQ,” [Online]. Available: https://quantumai.google/cirq. [Accessed 26 01 2021].

[26] D.-W. Systems, “Leap - The quantum cloud service,” [Online]. Available: https://www.dwavesys.com/take-leap. [Accessed 02 11 2020].

[27] P. Wittek, J. Biamonte, N. Pancotti, S. Lloyd, P. Rebentrost and N. Wiebe, “Quantum machine learning,” Nature, vol. 549, no. Nature 549, 195-202 (2017), p. 24, 2017.

[28] Amazon, “Braket,” Amazon, [Online]. Available: https://aws.amazon.com/braket/. [Accessed 12 01 2021].

[29] A. Hassidim, S. Lloyd and A. W. Harrow, “Quantum algorithm for solving linear systems of equations,” Phys. Rev., vol. 103, no. pp. 150502. Lett. vol. 15, no. 103 1079-7114, p. 15, 2008.

[30] G. Brassard, S. Gambs and E. Aïmeur, “Machine learning in a quantum world. In: Advances in Artificial Intelligence,” in Springer, . pp. 431–442, 2006.

[31] J. Goldstone, S. Gutmann and E. Farhi, “A quantum approximate optimization algorithm,” arXiv:1411.4028.

[32] P. Ronagh, J. S. Oberoi, D. Crawford, N. Ghadermarzy and A. Levit, “Reinforcement Learning Using Quantum Boltzmann Machines,” Quantum Information & Computation, Vols. Volume 18 (1-2), no. arXiv:1612.05695v3 , 2018.

[33] T. M. Barlow, A. Beige, A. C. Lewis and W. Huang, “Hidden Quantum Markov Models and Open Quantum Systems with Instantaneous Feedback,” Interdisciplinary Symposium on Complex Systems, Emergence, Complexity and Computation., no. arXiv:1406.5847v2, p. 10, 2014.

[34] P. Shor, “Algorithms for quantum computation: discrete logarithms and factoring,” IEEE Comput. Soc. Press., vol. Proceedings 35th Annual Symposium on Foundations of Computer Science., no. 10.1109/SFCS.1994.365700., p. 134, 1994.

[35] P. Khosla, S. Saini, G. Singh and M. Kaur, “Quantum Driven Machine Learning,” International Journal of Theoretical Physics, vol. 59, no. 4013–4024, p. 12, 2020.

[36] A. D. Córcoles, V. Havlíček, A. Kandala and K. Temme, “Supervised learning with quantum-enhanced feature spaces,” Nature, vol. 567, no. https://doi.org/10.1038/s41586-019-0980-2, 2019.