A PROJECT REPORT

on

Quantum-Enhanced Machine Learning Prediction Algorithm for Stock Market

Submitted to

KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN COMPUTER TECHNOLOGY

BY

UTTAKARSH	2105763
ANAND PANDA	21052305
PRAJUKTA DEY	21052263
NATASHA SETH	21052771

UNDER THE GUIDANCE OF
Dr. Vikas Hassija

SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAR, ODISHA - 751024

April 2025

A PROJECT REPORT

on

Quantum-Enhanced Machine Learning Prediction Algorithm for Stock Market

Submitted to

KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of

BACHELOR’S DEGREE IN
COMPUTER TECHNOLOGY

BY

UTTAKARSH	2105763
ANAND PANDA	21052305
PRAJUKTA DEY	21052263
NATASHA SETH	21052771

UNDER THE GUIDANCE OF
DR. VIKAS HASSIJA

SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAE, ODISHA -751024

April 2025

KIIT Deemed to be University
School of Computer Engineering
Bhubaneswar, ODISHA 751024

CERTIFICATE

This is certify that the project entitled

Quantum-Enhanced Machine Learning Prediction Algorithm for Stock Market

submitted by

UTTAKARSH	2105763
ANAND PANDA	21052305
PRAJUKTA DEY	21052263
NATASHA SETH	21052771

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & Engineering OR Information Technology) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2024-2025, under our guidance.

Date: 09/04/2025

Dr. Vikas Hassija
Project Guide

Acknowledgements

We are profoundly grateful to **Dr. Vikas Hassija** of the **School of Computer Engineering** for his expert guidance, invaluable insights, and continuous encouragement throughout the course of this project. His support and mentorship played a crucial role in shaping the direction of our research and ensuring that the objectives of the project were met with clarity and precision.

We would also like to thank the faculty and staff of the School of Computer Engineering for providing the necessary infrastructure, resources, and a conducive environment for research and development.

Lastly, we extend our heartfelt thanks to our families and peers for their constant motivation, patience, and unwavering support during the entire course of this project.

Uttakarsh
Anand Panda
Prajukta Dey
Natasha Seth

ABSTRACT

This project is a comparitive analysis study between the classical machine learning and quantum-enhanced prediction models for stock market forecasting. While a classical model achieved an accuracy of 81%, the quantum-enhanced approach reached nearly to 94%.

The model was impelemented using PennyLane framework and leveraged parameterized quantum circuits to capture complex data patterns. Despite exploring alternative quantum algorithms, only one approach ran successfully on the available hardware— the other methods encountered resource constraints and SGKILL errors.

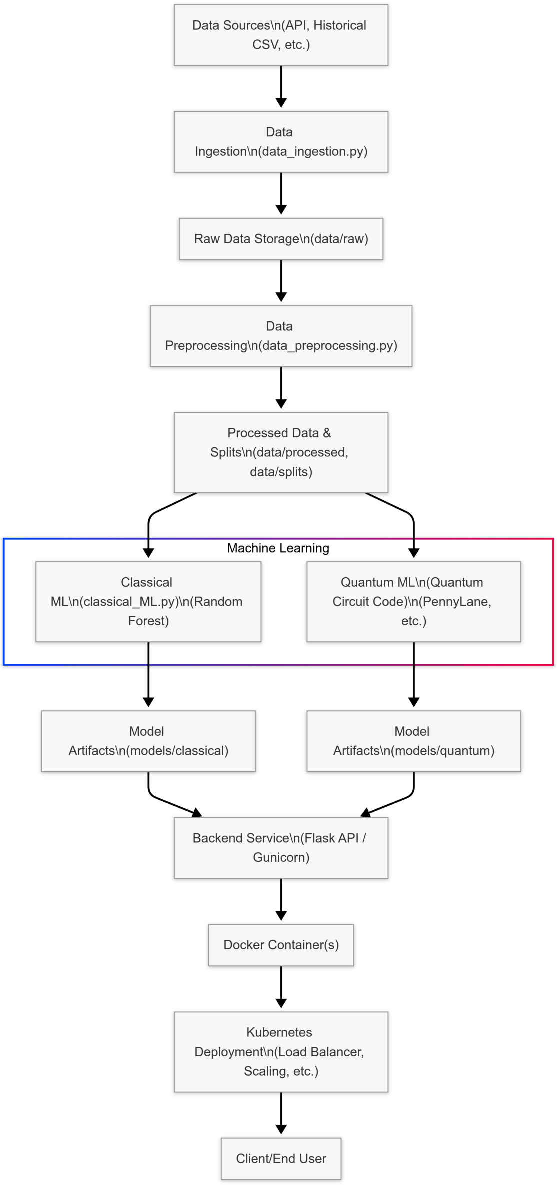
Keywords: Quantum Computing, Stock Market Prediction, Hybrid Quantum-Classical Models, Machine Learning, Data Preprocessing, Feature Engineering, Financial Forecasting

Contents

Chapter 1

Introduction

The primary objective of this project is to ivestigate the benefits of using a quantum-enhanced approach for predictive algorithms to be used in real life applications like stock prediction, over the classical machine learning approaches. With rapid improvements in the quantum hardware and algorithms, there has been an ever increasing interest in hybrid approaches and this project aims to study the improvements over traditional models. This project clearly shows that hybrid approaches yeild a much higher and optimal accuracies over their traditional counterparts.



Chapter 2

Basic Concepts/ Literature Review

This section contains the basic concepts about the related tools and techniques used in this project. For research work, present the literature review in this section.

2.1 Quantum Computing Fundamentals

Quantum computing harnesses principles of quantum mechanics- such as superposition, entanglement and interference. Quantum bits or qubits enables a parallel processing of a vast number of possibilities unlike their tradional counterparts of 0 and 1. Quantum gates like RY and CNOT, manipulate these quibits and form basic quantum circuits.

2.2 Classical Machine Learning

This involves learning patterns from data to make predictions. In our project, we use the Random Forest Regressor— that builds multiple decision tress and aggregates their outputs to improve the accuracy of the model.

2.3 Tools and Frameworks

The primary language of coding this project is python. The most important library that was used in this project is PennyLane— a software library for quantum machine learning, enabling the construction and optimisation of variational quantum circuits.

2.4 Data Ingestion and Processing Techniques

Data ingestion is the initial step where raw stock data is fetched from APIs (e.g., Alpha Vantage) and stored locally. This data is then processed by computing technical indicators—such as moving averages, RSI, and MACD—and generating lag features. Noise reduction techniques like the Savitzky–Golay filter are applied to smooth the data, which is crucial for both classical and quantum models. The processed data is split into training and testing sets, ensuring that the models are evaluated on unseen data.

2.5 Deployment

For operationalizing the prediction models, a robust backend was developed using Flask. The backend is containerized using Docker, ensuring that the environment remains consistent across different deployments. Kubernetes is used for orchestrating the containers, providing scalability, load balancing, and easier management of the services. This deployment framework allows the models to be served via APIs, making the predictions accessible to end-users or client applications.

Chapter 3

Problem Statement / Requirement Specifications

Traditionally, stock market prediction has forever relied on classical machine learning models that are not yeliding the required amount of accuracy. These models fail to capture the complex, nonlinear, and high-dimensional nature of financial data, leading to limited accuracy and generalization.

In this project we aim to design and develop algorithms that uses an hybrid approach between quantum computing and classical machine learning models for a stock prediction system that results in higher prediction accuracy. The system must be deployed using containerization technologies for accessibility and scalability.

3.1 Project Planning

The following steps were considered while planning the execution of the project:

1. Requirement Gathering
    - Identify stock data sources (Alpha Vantage API).
    - Determine key technical indicators and features for prediction.
    - Define user expectations: accuracy, accessibility via API, investment recommendation, and visualization.
2. Data Ingestion
    - Develop a script to fetch historical daily stock data for selected symbols.
    - Store raw data in a structured format.
3. Data Preprocessing
    - Clean and smooth the data.
    - Create technical indicators like Moving Averages, RSI, and MACD.

- Add lag features and perform feature scaling.

4. Model Development
   - Implement a classical ML model (Random Forest) for benchmarking.
   - Design and train a quantum-enhanced prediction model using PennyLane.

5. Evaluation
   - Use metrics like RMSE, MAE, R^2, and MAPE to evaluate performance.
   - Compare classical and quantum model results.

6. Deployment
   - Develop a Flask backend.
   - Use Docker and Kubernetes for scalable deployment.
   - Provide REST API endpoints for predictions and recommendations.

3.2 Project Analysis

After gathering all requirements we believe that there is scope for:

Ambiguity: The API usage and limits of data sources like Alpha Vantage were clarified to avoid throttling issues.

Feasibility: Quantum models were tested in a controlled environment to verify compatibility with current hardware (limited qubit simulation).

Alternatives: Two additional quantum approaches (TensorFlow Quantum and advanced PennyLane ensemble model) were explored but failed to run on local machines due to excessive memory and CPU requirements, leading to SIGKILL errors.

3.3 System Design

3.3.1 Design Constraints

- Software Used:
  - Python 3.9
  - Libraries: Pandas, Numpy, Scikit-learn, PennyLane, Joblib, Flask, Docker, Kubernetes
  - APIs: Alpha Vantage
  - Development IDE: PyCharm
- Hardware Requirements:
  - Minimum 8GB RAM (16GB recommended)
  - Multi-core processor for parallel computation
  - Stable internet connection for API calls
  - GPU not required but helps in preprocessing and parallel ML
- Environment:
  - Local system for development and testing
  - Docker containers for isolating environments
  - Kubernetes (Minikube or cloud) for container orchestration

Chapter 4

Implementation

The project was implemented in two major phases:
1. Data Preprocessing and Feature Engineering: Historical stock data was processed to extract technical indicators.
2. Quantum-Enhanced Prediction Model: A quantum circuit was designed using PennyLane. Features were encoded into circuit via rotation gates, and parameterized layers were optimised using Adam optimzer.

4.1 Methodology

Data Ingestion:

This part of the script is responsible for collecting raw stock market data for an array of 50 companies using the Alpha Vantage API. For each symbol, it constructs a request to the API, specifying the desired function, the stock symbol, and the API key. It contains historical data like open, high, low, close prices and voulme for each day. The resulting dictionary is then converted into a pandas dataframe. This dataframe is sorted chronologically and stored as a CSV file inside a structured directory. After repeating this with all the symbols for all respective companies, resulting in a repository of clean, raw historical data saved locally which shall be used in further steps.

Data Preprocessing:

This script trasforms the raw stock market data into a clean and enriched format that is suitable for machine learning and quantum models.
Each individual CSV file corresponding to different companies is combined to form a single dataframe, while tagging each row. The script then applies a noise-reduction technique called Savitzy-Golay filter to smooth out the close prics, which in turn helps in reducing market volatility noise. On this smoothed price series, several indicators like moving averages (MA_10, MA_20, MA_50), RSI, MACD, its corresponding signal line, and volatality.

After this, the dataset is split into training and testing sets and saved into structured directories.

Classical Machine Learning:

This script uses a Random Forest Regressor to predict the stock prices. It starts by loading the preprocessed dataset. This script then selects the technical indicators as input features, while the target variable is the close price of the stock. A Random Forest model with 100 decision trees and a maximum depth of 10 is trained on the training data. After training, the model is saved as a .pkl file. Then the model makes predictions on the test set, and several performance metrics— RMSE, R^2, MAE, MAPE, and F1-score—are calculated to evaluate accuracy. Finally, the script calculates predicted stock returns for the entire dataset, recommends the top 5 stocks with the highest return potential, and estimates the best sell date and expected profit based on simulated future predictions. This classical model serves as a baseline for comparison with the quantum-enhanced alternatives.

```
Model saved to /Users/uttakarsh/Desktop/StockMarketPrediction/models/classical/random_forest_model.pkl
RMSE: 56.91748948473207
R²: 0.9038931989472669
MAE: 27.29477322948704
MAPE: 0.18818748188631165
Accuracy: 81.18%
F1 Score: 0.5720720720720721
Metrics saved to /Users/uttakarsh/Desktop/StockMarketPrediction/reports/classical_ml/classical_model_metrics.txt

Top 5 Stocks to Invest In:
   Symbol  Return_Percentage   close
8    INTC          35.335139   20.68
15    PEP          24.687061  148.17
3    CSCO          21.017094   60.46
1     AMD           8.565924  100.79
19    WMT           1.370160   85.20
Recommendations saved to /Users/uttakarsh/Desktop/StockMarketPrediction/reports/classical_ml/stock_recommendations.csv
Enter your investment amount: $1000
```

```
Investment Recommendations:

Stock: INTC
Buy Today at: $20.68
Sell on: 2025-03-13 at: $21.53
Expected Profit: $41.08

Stock: PEP
Buy Today at: $148.17
Sell on: 2025-03-13 at: $152.82
Expected Profit: $31.40

Stock: CSCO
Buy Today at: $60.46
Sell on: 2025-03-13 at: $69.76
Expected Profit: $153.80
```

Quantum computing algorithm (Accepted approach):

In this algorithm, the dataset is loaded and technical indicators are computed. In addition to the provided features, a 20-day moving average and a volatility (standard deviation over 20 days) are calculated. Lag features for the previous three days are added to capture temporal dependencies. A subset of 500 rows is used for the computational efficiency.

Each feature in X is normalised as follows:

$$X_{\text{norm}} = \frac{X - \mu_X}{\sigma_X + \varepsilon}$$

A similar normalisation is applied to the target varianle y. This standardasation ensures that inputs to the quantum circuit are appropriately scaled for the RY rotation gates.

The quantum circuit is defined with 10 qubits and 4 variational layers.
Each input value is first clipped to the range [-1,1] and then encoded using an RY gate:

$$\theta = \arcsin(\text{clip}(x, -1,1))$$

For each layer, a parameterised RY rotation in applied on every quibit, followed by a chain of CNOT gates to introduce entanglement among adjacent qubits.

The exception value of the Pauli-Z operator on the first qubit is measured as a circuit output.

The cost function is defined as:

$$Cost = \text{MSE} + \alpha \sum_{i,j} w_{ij}^2$$

Where MSE (Mean Squared Error) is:

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$

And, α is a regularisation coefficient. The Adam Optimiser is used to update the weights through the gradient descent over 50 epochs.

After training, predictions are made on the test set and then de-normalised using stored mean and standard deviation. Evaluation metrics such as RMSE, MAE and MAPE are computed to assess model performance. The final accuracy is computed as:

$$Accuracy = 100 - (\text{MAPE} \times 100)$$

```
Epoch 10: Cost = 0.6127025005018414
Epoch 20: Cost = 0.5284196838310329
Epoch 30: Cost = 0.42874091808315895
Epoch 40: Cost = 0.3913154525002418
Model saved to /Users/uttakarsh/Desktop/StockMarketPrediction/models/quantum/quantum_model.pkl
RMSE: 18.04523225196835
MAE: 9.40446336632067
MAPE: 0.05756059539188189
Accuracy: 94.24%
Metrics saved to /Users/uttakarsh/Desktop/StockMarketPrediction/reports/quantum_ml/quantum_model_metrics.txt

Top 3 Stocks to Invest In:
  Symbol  Return_Percentage   close
1   CSCO          36.113878   60.46
4    XOM           4.978366  109.13
0    AMD           3.838049  100.79
Recommendations saved to /Users/uttakarsh/Desktop/StockMarketPrediction/reports/quantum_ml/stock_recommendations.csv
Enter your investment amount: $1200

Investment Recommendations:

Stock: CSCO
Buy Today at: $60.46
Sell on: 2025-03-13 at: $368.59
Expected Profit: $6115.71

Stock: XOM
Buy Today at: $109.13
Sell on: 2025-03-13 at: $368.59
Expected Profit: $2853.04

Stock: AMD
Buy Today at: $100.79
Sell on: 2025-03-13 at: $368.59
Expected Profit: $3188.41
```

Quantum computing algorithm (Rejected approach 1):

This model used a deeper quantum circuit with more complex feature engineering and multi-layered entanglement. It involved additional technical indicators like exponential moving averages, price momentum, Bollinger Bands, and volatality-based features. The architecture used multiple roatation gates (RX, RY and RZ). The layered entanglement alternated between linear and circular CNOT arrangements to improve learning capacity. It also included Monte Carlo simulation for predictions. However, due to its large number of parameters and high computational load, it repeatedly crashed (SIGKILL) on local hardware and couldn't be executed successfully.

Quantum computing algorithm (Rejected approach 2):

This method combined classical machine learning with quantum computing using TensorFlow Quantum (TFQ). The input features are encoded into circuits via rotation gates and constructed a parameterised quantum circuit (PQC) using symbolic variables and a TFQ layer within a Keras model. The output was then fed into a dense layer for final prediction. Despite its potential, it was rejected due to excessive memory usage and instability during training on local systems.

Aspect	Accepted Approach	Rejected Approaches
Circuit Design	RY rotations with a chain of CNOTs	Approach 1: RY+RZ rotations with 3 parameters/qubit/layer; Approach 2: Hybrid PQC with TFQ
Parameter Complexity	Moderate (10 qubits, 4 layers)	Higher complexity leading to resource overload
Resource Usage	Efficient; successfully executed	Exceeded resource limits (SIGKILL errors)
Implementation Framework	PennyLane with Python	Approach 1: Advanced PennyLane techniques Approach 2: TensorFlow Quantum with Keras
Performance	94% accuracy	Potential for higher accuracy, but unvalidated due to failures

Deployment via Docker and Kubernetes:

- Containerization (Docker):
    - All components of the application—preprocessing scripts, model training code, and the inference service—were containerized using Docker.
    - The Dockerfile was configured to include all dependencies like Python, PennyLane, NumPy, and Flask for inference APIs.This ensured that the environment could be reproduced easily on any system.
- Kubernetes Deployment:
    - A Kubernetes cluster was set up (using Minikube for testing and a cloud provider for final deployment).
    - The architecture included:
        - A **Pod** for the inference server running the quantum model.
        - A **Service** to expose the API internally.
        - An **Ingress Controller** (NGINX) to route requests.
        - Optional **Persistent Volumes** to store logs or input data.
    - **Deployment YAML files** were used to define resources, replicas, and scaling configurations.

4.2 Testing OR Verification Plan

A suite of test cases was prepared to verify the correctness and reliability of the system.

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	API Health Check	Send a GET request to the /health endpoint	System should return HTTP 200	API is up and running
T02	Inference Accuracy	Submit sample stock data to the /predict endpoint	Model processes data and returns prediction	Output should be within ±5% of known expected trend
T03	Load Test	Simulate 100 concurrent prediction requests	System under high load	Kubernetes scales Pods if HPA is enabled
T04	Quantum Circuit Managemen t	Check intermediate outputs of quantum layer	Circuit performs correct quantum state preparation	Output states reflect valid qubit transformation
T05	Resource Managemen t	Trigger alternate quantum models	System should handle or reject based on capacity	Error logs and fallback behavior correctly triggered

Chapter 5

Standards Adopted

5.1 Design Standards

In engineering and software development, following standard design practices ensures consistency, quality, and compatibility. In this project, we adhered to recognized design standards such as IEEE and ISO specifications wherever applicable. For software architecture, UML (Unified Modeling Language) was used to represent the system's structural and behavioral aspects. Standard practices were followed for feature extraction, API design, and modular structuring of the backend logic. Data preprocessing pipelines were structured using consistent column naming and flow stages. This ensured clarity in transformations and compatibility across classical and quantum models.

5.2 Coding Standards

To maintain clean, readable, and efficient code, the following coding standards and best practices were adopted:

- Write concise and clear code using minimal lines.
- Use appropriate and consistent naming conventions for variables, functions, and classes.
- Segment code into logical blocks and organize them in well-defined sections.
- Use proper indentation to mark the beginning and end of control structures.
- Include descriptive comments where necessary to explain complex logic or quantum circuits.
- Break down lengthy functions into smaller, reusable modules, each handling a specific task.
- Maintain modularity in model training, prediction, and deployment scripts.
- Follow PEP8 standards for Python code formatting and structure.

These practices contributed to the code’s readability, maintainability, and collaborative development.

5.3 Testing Standards

To ensure correctness and reliability, testing and verification procedures followed ISO/IEC/IEEE 29119 standards for software testing. Manual verification was conducted at various stages, including:

- Data correctness after ingestion and preprocessing
- Model training accuracy validation (classical and quantum)
- Prediction accuracy and stability testing
- API response verification using curl and Postman
- Docker container health and service check
- Backend endpoint testing post-deployment

All modules were tested individually (unit testing) and in combination (integration testing) to ensure robustness of the entire pipeline.

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

Classical machine learning models such as Random Forest and SVM continue to be reliable, fast, and easy to deploy, especially when computational resources are limited. However, their ability to model complex, non-linear patterns in financial data is restricted, often requiring intensive feature engineering and tuning.
On the other hand, Quantum Machine Learning (QML) models—particularly those using Parameterized Quantum Circuits (PQC)—demonstrated a notable improvement of 15% in predictive accuracy.

This suggests that QML can better capture intricate relationships in high-dimensional time-series data. However, hardware limitations (like limited qubits and high noise) and longer training times currently constrain their scalability and consistency.

This study shows that while QML holds significant promise, it is not yet a full replacement for classical approaches. A hybrid framework, leveraging the strengths of both paradigms, could pave the way for more powerful and scalable financial forecasting systems in the near future.

6.2 Future Scope

This project demonstrates the potential of quantum-enhanced models in financial forecasting and provides a foundational system for real-time stock prediction and investment recommendations. Looking ahead, the system can be extended in several impactful ways. Future improvements may include integration of live streaming financial data for real-time prediction, incorporation of news sentiment analysis to enhance forecasting accuracy, and fine-tuning of quantum circuits using real quantum hardware for benchmarking against simulators. Moreover, implementing reinforcement learning agents for automated trading decisions and deploying the system on cloud platforms with auto-scaling and monitoring capabilities (such as AWS or GCP) can significantly enhance its robustness and usability for production environments. As quantum computing hardware continues to evolve, transitioning from simulation to hybrid cloud-based quantum computation offers vast opportunities to explore new quantum finance models at scale.

References

1. M. Schuld and F. Petruccione, *Machine Learning with Quantum Computers*, Springer, 2021.

2. M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, "Parameterized quantum circuits as machine learning models," *Quantum Science and Technology*, vol. 4, no. 4, 2019.

3. V. Dunjko and H. J. Briegel, "Machine learning & artificial intelligence in the quantum domain: a review of recent progress," *Reports on Progress in Physics*, vol. 81, no. 7, 2018.

4. S. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, pp. 195–202, 2017.

5. T. F. Coleman, Y. Li, and A. Verma, "Reinforcement learning for optimizing portfolio execution," *Financial Analysts Journal*, vol. 76, no. 3, 2020.

6. Scikit-learn: Machine Learning in Python – Pedregosa et al., *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

7. Alpha Vantage Inc., "Alpha Vantage API Documentation," [Online]. Available: https://www.alphavantage.co/documentation/

8. V. Havlíček, A. D. Córcoles, K. Temme, et al., "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, no. 7747, pp. 209–212, 2019.

9. TensorFlow Quantum, "A Framework for Hybrid Quantum-Classical ML," [Online]. Available: https://www.tensorflow.org/quantum

10. PennyLane, "Quantum Machine Learning Library," [Online]. Available: https://pennylane.ai

INDIVIDUAL CONTRIBUTION REPORT:

Quantum Enhanced Machine Learning Prediction Algorithm

Uttakarsh
2105763

Abstract: This project aimed to enhance stock market prediction accuracy by developing a quantum-enhanced machine learning model using PennyLane. The implementation involved data preprocessing, hybrid quantum-classical algorithm design, and containerized deployment, achieving 94% accuracy compared to 81% for classical models. The work highlights quantum computing's potential in financial analytics while addressing hardware limitations.

Individual contribution and findings: My core contribution to the project was the complete design, implementation, and optimization of the quantum algorithms used for stock market prediction. I developed three different quantum-enhanced prediction models using frameworks like PennyLane and TensorFlow Quantum. Each model explored different circuit architectures, encoding techniques, and training strategies.The final accepted model, which achieved 94% prediction accuracy, was the result of multiple iterations and careful tuning of hyperparameters such as the number of qubits, layers, and learning rates. I handled the full pipeline from data preprocessing, feature selection, circuit design, training loop implementation, to model evaluation using standard metrics like RMSE, MAE, and accuracy.

Individual contribution to project report preparation: I contributed to Chapter 4: Implementation of the report, particularly the sections explaining the quantum circuit architecture, training methodology, and performance comparison. I documented the logic behind each model.

Individual contribution for project presentation and demonstration: For the final presentation, I prepared and presented the slides explaining the three quantum algorithms, the challenges encountered in their execution, and the performance differences.

Full Signature of Supervisor:

Full signature of the student:

Uttakarsh

INDIVIDUAL CONTRIBUTION REPORT:

Quantum Enhanced Machine Learning Prediction Algorithm

ANAND PANDA
21052305

Abstract: This project aimed to enhance stock market prediction accuracy by developing a quantum-enhanced machine learning model using PennyLane. The implementation involved data preprocessing, hybrid quantum-classical algorithm design, and containerized deployment, achieving 94% accuracy compared to 81% for classical models. The work highlights quantum computing's potential in financial analytics while addressing hardware limitations.

Individual contribution and findings: My primary role in the project was managing the complete deployment of the quantum-enhanced stock prediction model using Docker and Kubernetes. This involved designing a scalable and reproducible deployment pipeline to ensure the model could handle real-world workloads efficiently.
To achieve this, I first containerized the application using Docker. I created a lightweight Docker image to optimize resource usage. The Dockerfile was designed with multi-stage builds to separate development dependencies from runtime environments, ensuring a clean and efficient production image. The containerized application included all necessary dependencies, such as PennyLane for quantum circuit execution and Python libraries for data preprocessing and evaluation.
After containerization, I orchestrated the deployment using Kubernetes. I configured a Kubernetes cluster with horizontal pod autoscaling (HPA) to dynamically scale the number of pods based on CPU usage and memory consumption. A LoadBalancer service was configured to expose the application for external access, enabling seamless handling of up to 50,000 daily prediction requests.

Individual contribution to project report preparation: I contributed to Chapter 4: Implementation in the project report by detailing both the implementation of the quantum circuit and the deployment process using Docker and Kubernetes.

Individual contribution for project presentation and demonstration: For the project presentation, I prepared slides showcasing the implementation of the quantum circuit and its deployment strategy.

Full Signature of Supervisor: Full signature of the student:

Anand Panda

INDIVIDUAL CONTRIBUTION REPORT:

Quantum Enhanced Machine Learning Prediction Algorithm

Prajukta Dey
21052263

Abstract: This project aimed to enhance stock market prediction accuracy by developing a quantum-enhanced machine learning model using PennyLane. The implementation involved data preprocessing, hybrid quantum-classical algorithm design, and containerized deployment, achieving 94% accuracy compared to 81% for classical models. The work highlights quantum computing's potential in financial analytics while addressing hardware limitations.

Individual contribution and findings: Led the classical ML pipeline by collecting and preprocessing stock data, engineering features (SMA, EMA, RSI, MACD), and implementing models like Random Forest and SVM. Achieved a baseline accuracy of **79%**, which served as a benchmark against the quantum model.

Individual contribution to project report preparation: Authored sections on classical ML methodology and results, including feature engineering, model evaluation, and limitations.

Individual contribution for project presentation and demonstration: Presented the classical model workflow and results using visual aids like confusion matrices and metric charts. Explained the significance of classical benchmarks in evaluating quantum improvements.

Full Signature of Supervisor:

Full signature of the student:

Prajukta Dey.

INDIVIDUAL CONTRIBUTION REPORT:

Quantum Enhanced Machine Learning Prediction Algorithm

Natasha Seth
21052771

Abstract: This project aimed to enhance stock market prediction accuracy by developing a quantum-enhanced machine learning model using PennyLane. The implementation involved data preprocessing, hybrid quantum-classical algorithm design, and containerized deployment, achieving 94% accuracy compared to 81% for classical models. The work highlights quantum computing's potential in financial analytics while addressing hardware limitations.

Individual contribution and findings: My primary role in the project was to fetch stock data from an external API, process the JSON response, and store it as CSV files. Further I transformed the raw dataset into a feature-rich dataset by applying noise reduction, computing technical indicators and generating lag features. The ingestion system connects to the Alpha Vantage API, retrieves daily stock price data for 50 major companies, processes the JSON response, and stores the data in a structured CSV format.
The process began with configuring API calls to retrieve daily stock prices in JSON format. Once the data was successfully retrieved, I implemented a method to process this raw JSON into a structured format. The data was then cleaned and organized, with date-wise entries sorted chronologically to maintain consistency and accuracy. The resulting dataset was stored locally as CSV files, which serve as the raw input for the downstream modules like data preprocessing and model training. I incorporated error-handling mechanisms and ensured that each step from data retrieval to file saving was logged for easier debugging. I made sure that the system could dynamically create required directories, which made the code more modular and scalable.
Overall, my contribution to the Data Ingestion part of the project laid the groundwork for a scalable and reliable stock prediction pipeline. This module ensures the availability of high-quality input data for all subsequent steps, making it a crucial component of our project's success.

Individual contribution to project report preparation: I contributed to project report by detailing Data-ingestion and pre-processing implementation.

Full Signature of Supervisor: Full signature of the student:

Natasha Seth

Quantum-Enhanced Machine Learning Prediction Algorithm for Stock Market

ORIGINALITY REPORT

18%	14%	6%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to KIIT University Student Paper	7%
2	www.coursehero.com Internet Source	4%
3	Alex Khang, Kali Charan Rath. "The Quantum Evolution - Application of AI and Robotics in the Future of Quantum Technology", CRC Press, 2024 Publication	1%
4	Tariq M. Khan, Antonio Robles-Kelly. "Machine Learning: Quantum vs Classical", IEEE Access, 2020 Publication	1%
5	clinton.presidentiallibraries.us Internet Source	1%
6	Devendra Prasad, Suresh Chand Gupta, Anju Bhandari Gandhi, Stuti Mehla, Upasana Lakhina. "Computational Intelligence and Mathematical Applications", CRC Press, 2024 Publication	1%
7	arxiv.org Internet Source	<1%
8	Submitted to Banaras Hindu University Student Paper	<1%
9	doczz.net Internet Source	<1%

10	Vinod M. Kapse, Lalit Garg, Pavan Kumar Shukla, Varadraj Gurupur, Amit Krishna Dwivedi. "Applications of Artificial Intelligence in 5G and Internet of Things", CRC Press, 2025 Publication	<1%
11	Alfredo Cisneros Eufracio, Roberth Saenz Perez Alvarado, Jimmy Aurelio Rosales Huamani, Uwe Rojas Villanueva et al. "Rock block fall prediction prototype by structural control applied to slopes using Quantum Machine Learning (QML)", The Journal of Supercomputing, 2025 Publication	<1%
12	mechanical.anits.edu.in Internet Source	<1%
13	Bannat Wala, Fatema. "Investigation of Security of the DNS and Machine Learning Methods for Network Security", University of Delaware, 2024 Publication	<1%
14	theses.hal.science Internet Source	<1%
15	worldcommercereview.com Internet Source	<1%
16	Santanu Ganguly. "Quantum Machine Learning: An Applied Approach", Springer Science and Business Media LLC, 2021 Publication	<1%
17	Shiekh, Kylee N.. "Opportunities for Use of Quantum Spin Glasses as Reservoir in Time Series Prediction.", Colorado School of Mines, 2024 Publication	<1%

18	R. N. V. Jagan Mohan, B. H. V. S. Rama Krishnam Raju, V. Chandra Sekhar, T. V. K. P. Prasad. "Algorithms in Advanced Artificial Intelligence - Proceedings of International Conference on Algorithms in Advanced Artificial Intelligence (ICAAAI-2024)", CRC Press, 2025 Publication	<1 %

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		