A Hybrid Quantum-Classical Fusion for Deep Semantic Paraphrase Detection

Devanarayanan K and **Fayas S Mohamad** and **Dheeraj V Mohan** and **Reshma Sheik**
TKM College of Engineering, Kollam, Kerala.
devanarayanank.2003@gmail.com, rezmasheik@gmail.com

Abstract

Paraphrase Detection is a core task in natural language processing (NLP) that aims to determine whether two sentences convey equivalent meanings. This work proposes a hybrid quantum–classical framework that integrates Sentence-BERT embeddings, simulated quantum feature encoding, and classical machine learning models to enhance semantic similarity detection. Initially, sentence pairs are embedded using Sentence-BERT and standardized through feature scaling. These representations are then transformed via rotation-based quantum circuits to capture higher-order feature interactions and non-linear dependencies. The resulting hybrid feature space, combining classical and quantum-inspired components are evaluated using LightGBM and deep neural network classifiers. Experimental results shows that the hybrid model incorporating quantum-inspired features achieved superior classification performance, yielding a 10% improvement in overall accuracy outperforming standalone deep learning baselines. These findings demonstrate that quantum–classical fusion enhances semantic feature extraction and significantly improves paraphrase detection performance.

1 Introduction

Recent studies have explored the intersection of quantum computing and Natural Language Processing (NLP) to enhance semantic understanding and text similarity modeling. Paraphrase detection is an important task in natural language processing that aims to identify whether two sentences convey the same meaning. It has applications in areas such as question answering, plagiarism detection, and semantic search (Madaan et al., 2016). Classical machine learning methods have achieved significant progress using embedding models and gradient boosting techniques. However, capturing deeper semantic relationships between sentence pairs remains a challenge due to the limitations of classical representations. Quantum computing provides an exciting way through the encoding of linguistic information into quantum states, which have a natural way to represent and process the correlations that are hard to model in classical environments. Emerging advances in Quantum NLP (Q-NLP) demonstrate that quantum circuits can represent structural and semantic relationships between sentence parts in manners that complement classical neural architectures (Meichanetzidis et al., 2023).

Earlier works (Buhrman et al., 2001) introduced quantum fingerprinting, demonstrating how quantum states can represent compact data signatures for efficient comparison—laying the theoretical foundation for quantum information comparison techniques. (Darwish et al., 2023) proposed a quantum genetic algorithm for semantic textual similarity estimation in plagiarism detection, highlighting quantum-enhanced optimization in NLP tasks. Gao et al. (Gao et al., 2024) developed a quantum-inspired hierarchical semantic interaction model for text classification that captures multi-level contextual relations between words. Meanwhile, Guarasci et al. (Guarasci et al., 2022) discussed the broader challenges and opportunities in quantum natural language processing, emphasizing scalability, noise resilience, and quantum circuit design constraints. In contrast, the present research focuses specifically on paraphrase detection using a hybrid quantum–classical framework, integrating both classical semantic embeddings and quantum circuit-based similarity estimation for more accurate and interpretable detection of paraphrased sentences. Due to the exponential cost of simulating larger circuits, the initial system encodes a low-dimensional subset of SBERT features into a 4-qubit circuit as a feasibility study. This establishes a baseline for scaling quantum components in future work.

The remainder of this paper is organized as

Proceedings of the QuantumNLP: Integrating Quantum Computing with Natural Language Processing, pages 20–25
November 24, 2025

follows: Section 2 defines the paraphrase detection task and describes the dataset used in this study. Section 3 outlines the proposed hybrid quantum–classical framework, including data preprocessing, SBERT embeddings, quantum feature generation, and classifier design. Section 4 reports the experimental results and performance analysis, while Section 5 concludes the paper.

2 Task and Dataset

The task of paraphrase detection can be defined formally as follows: given two input sentences *s1* and *s2*, decide if they are semantically equivalent. Although certain pairs can be determined by direct word overlap, most need more in-depth modeling of sentence structure, context, and meaning. The core problem is to identify semantic similarity that goes beyond surface-level patterns of words. This study used a supervised Kaggle dataset, equivalent to the Microsoft Research Paraphrase Corpus (MRPC) (Dolan and Brockett, 2005), which consists of paired text samples divided into training and test sets. The training set contains 3,554 sentence pairs, while the test set has 1,465 pairs. Each record in the dataset is structured with five columns: two identifier fields (#1 ID, #2 ID), two text fields (#1 String, #2 String), and one label (Quality) indicating the relationship between the sentences. The label indicates whether the two sentences are paraphrased (1) or not (0). The data set is in CSV format, encoded in UTF-8, and uses consistent delimiters for easy integration into machine learning models. Its structure allows experiments in sentence-level detection using classical, deep learning, or hybrid quantum (Biamonte et al., 2017) deep learning approaches. Using MRPC ensures standard benchmarking and comparability with prior NLP research.[1]

3 Methodology

This work explores a hybrid quantum-classical approach for paraphrase detection. The model first encodes each pair of sentences using Sentence-BERT (Reimers and Gurevych, 2019) to obtain dense vector embeddings. The hybrid representation—combining both classical embeddings and quantum features—is reduced in dimensionality using Principal Component Analysis (PCA) (Jolliffe, 2002). Two main classifiers are then applied: LightGBM for boosted decision-tree learning and a Multi-Layer Perceptron (MLP) for deep learning inference. The final output is a binary prediction indicating whether the input sentences are paraphrased (1) or not (0).

Figure 1 illustrates the proposed hybrid quantum–classical system architecture, which is organized into four primary layers, each contributing to efficient paraphrase detection through integrated quantum–classical processing (Havlíček et al., 2019). The Data Preprocessing Layer is responsible for acquiring, cleaning, and organizing the input data. It pairs sentences with their corresponding labels, removes missing or noisy entries, ensures balanced class distribution, and stores the cleaned data along with their embeddings for subsequent processing. The Embedding Layer uses SentenceBERT that transforms textual data into dense numerical representations using sentence-level embedding models, capturing the semantic relationships necessary for downstream learning. Each of them acts as a parameter for the learning and testing stages. The features are balanced by standardization for fast convergence, so that each parameter has mean 0 and standard deviation 1. The Hybrid Processing Layer augments these classical embeddings with quantum-enhanced representations to capture higher-order dependencies and improve discriminative capability. This layer integrates modules for quantum feature generation, dimensionality reduction using PCA, and feature fusion to form a unified hybrid feature space. Finally, the Learning and Prediction Layer manages model training and inference, leveraging both classical and hybrid machine learning models to perform paraphrase classification as a binary classification task.

3.1 Data Preprocessing

The preprocessing stage begins by identifying the text and label columns in the dataset. Rows with missing values in these columns are removed to maintain data consistency. The text columns are cast to string type, and the label column to integer type. Sentence pairs are then constructed by concatenating the two text columns with a separator token. For embedding generation, SBERT encodes each sentence pair into dense numeric vectors, which are standardized using a StandardScaler to achieve zero mean and unit variance. These standardized embeddings are subsequently used for

[1] https://www.kaggle.com/datasets/doctri/microsoft-research-paraphrase-corpus

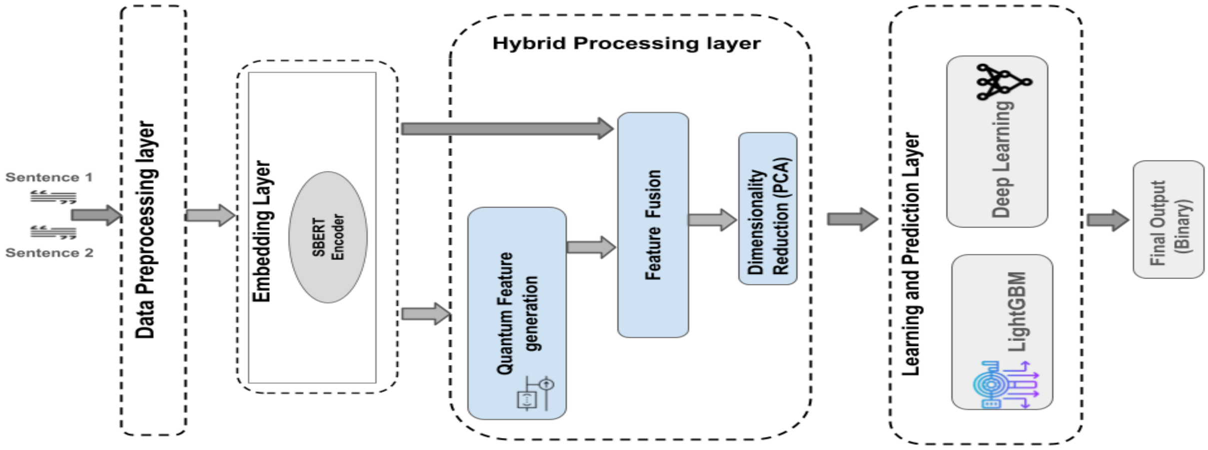

Figure 1: Hybrid Quantum-Classical Architecture for Paraphrase Detection

quantum feature generation. No additional text normalization, such as lowercasing or punctuation removal, is applied.

3.2 Quantum Feature Generation

The quantum feature generation process begins with the SBERT embeddings of sentence pairs. In the current implementation, four qubits are employed. The first four parameters of the 384-dimensional embedding vector are selected for quantum encoding.The first four SBERT dimensions were selected to ensure deterministic, reproducible mapping for a cost-feasible simulation. Future work will incorporate feature-selection methods (filter/wrapper techniques) to identify more discriminative embedding dimensions for quantum encoding. Each parameter is normalized to the range $[-\pi, \pi]$ and mapped to a qubit using an R_y rotation gate. The individual qubit states are combined using the Kronecker product to form a multi-qubit quantum state. Entanglement is introduced through a chain of CNOT gates connecting qubit 0 to 1, 1 to 2, and 2 to 3, thereby capturing correlations among qubits.CNOT gates introduce entanglement among the encoded parameters, allowing the measurement distribution to capture interaction effects beyond linear SBERT encoding. These interactions contribute to the hybrid feature space's expressiveness. The resulting quantum state is measured, yielding two measurement values per qubit, for a total of sixteen output values. These outputs constitute the quantum feature vector, which is then used as input to the hybrid model alongside the SBERT embeddings. Figure 2 illustrates the quantum circuit for the hybrid models, where four qubits are initialized with R_Y rotations, entangled via CNOT gates. Only a 4-dimensional slice of the SBERT vector is used for quantum encoding, as compact encoding is designed as a nonlinear feature transformation rather than a full high-dimensional quantum embedding. The circuit is fixed and non-trainable, with parameters directly mapped from SBERT values.

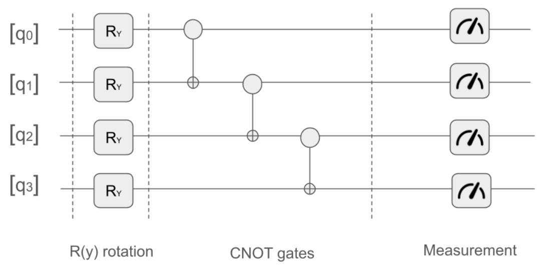

Figure 2: Quantum Circuit

3.3 Classifier Models

This section presents three model variants used in our study for paraphrase detection: (1) a LightGBM (LGBM) classifier based on gradient boosting with optimized hyperparameters; (2) a deep learning (DL) model comprising three fully connected layers with ReLU activations and dropout regularization; and (3) a hybrid model that integrates SBERT embeddings with quantum-inspired features, used in conjunction with either the LGBM or DL classifier to exploit both classical and quantum representations.

3.3.1 LGBM

LightGBM (Ke et al., 2017) is a gradient boosting framework that builds ensembles of decision trees sequentially, with each tree aiming to correct the residual errors of its predecessors. The model is tuned using a hyperparameter grid with the number of leaves 31, 63, 127, 255 and learning rates 0.01, 0.05, 0.1, over a maximum of 1000 boosting iterations. SBERT embeddings of sentence pairs serve as input features, and the best combination of leaves and learning rate is selected as the final model. Figure 3 visually explains the "leaf-wise" growth strategy of the LGBM algorithm. Instead of growing level by level, the tree is built by expanding the leaf that will cause the largest reduction in error.

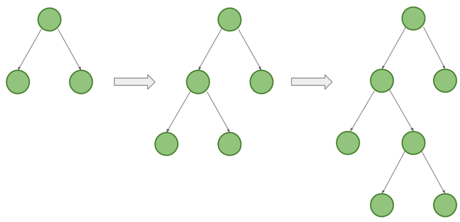

Figure 3: Growth strategy of the LGBM algorithm

3.3.2 DL

We employ a fully connected feedforward neural network for classification. The model takes SBERT embeddings as input and passes them through three hidden layers of 2048, 1024, and 512 neurons with ReLU activations and 0.3 dropout. It is trained for 50 epochs using the Adam (Kingma, 2014) optimizer and cross-entropy loss with a batch size of 128. Figure 4 depicts the Deep Learning model used in this study. It is a fully connected feedforward network comprising multiple hidden layers with ReLU activations, each followed by dropout for regularization, and a Softmax output layer for classification.

3.3.3 Hybrid Model

The hybrid model combines SBERT embeddings with quantum-inspired features to enhance the learning. The SBERT embeddings and quantum features are concatenated to form a hybrid feature vector. This hybrid representation serves as input to either a hybrid LGBM classifier or a hybrid DL model.

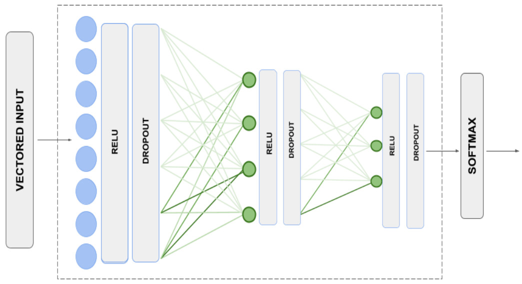

Figure 4: Deep Learning Model Architecture

4 Results & Discussions

Table 1 shows the performance metrics of the models in different sizes of training data. In terms of accuracy, the Hybrid-LGBM model achieved the highest overall performance, reaching 0.69 in 70% of the data. It consistently outperformed all other models across the 20–80% data range. The DL and Hybrid-DL models lagged behind the LGBM and Hybrid-LGBM models, likely due to overfitting in the DL models. The performance of Hybrid-DL improves significantly at higher data percentages (80–100%). In precision, the Hybrid-LGBM again outperformed the other models, achieving its best precision at 70% of the data. The DL and Hybrid-DL models initially showed lower precision, but the Hybrid-DL model steadily improved after 60% of the data, ultimately achieving its highest precision at 100% of the dataset. Recall indicates that LightGBM and Hybrid-LGBM perform best, both clearly outperforming DL and Hybrid-DL, with Hybrid-LGBM showing slightly higher recall than LGBM. Hybrid-DL improves notably after 60% of the data, surpassing DL at 80% and peaking at 100%. While both DL models perform similarly on smaller datasets, Hybrid-DL achieves higher F1 scores at larger data sizes (80–100%), outperforming the other models. At 100 percent training data, the Hybrid-LGBM model exhibited higher accuracy but a lower F1 score due to dataset class imbalance. LGBM optimizes leaf-wise splits that increase precision at the cost of recall, which impacts the F1 metric. The best balance between accuracy and F1 was observed in 70% training data. The Hybrid-DL model showed sensitivity to overfitting due to the larger hybrid feature dimension. Additional regularization and smaller architectures will be explored in future phases.

%	Model	Accuracy	Precision	Recall	F1-Score
10	LGBM	0.6767	0.6559	0.6767	0.6228
	DL	0.6528	0.6270	0.6528	0.6281
	Hybrid-LGBM	0.6685	0.6440	0.6685	0.5998
	Hybrid-DL	0.6432	0.6262	0.6432	0.6309
20	LGBM	0.6828	0.6671	0.6828	0.6297
	DL	0.6576	0.6348	0.6576	0.6365
	Hybrid-LGBM	0.6849	0.6868	0.6849	0.6142
	Hybrid-DL	0.6411	0.6297	0.6411	0.6338
30	LGBM	0.6842	0.6659	0.6842	0.6403
	DL	0.6336	0.6226	0.6336	0.6268
	Hybrid-LGBM	0.6863	0.6757	0.6863	0.6299
	Hybrid-DL	0.6528	0.6281	0.6528	0.6299
40	LGBM	0.6808	0.6626	0.6808	0.6293
	DL	0.6514	0.6362	0.6514	0.6405
	Hybrid-LGBM	0.6883	0.6794	0.6883	0.6326
	Hybrid-DL	0.6377	0.6262	0.6377	0.6305
50	LGBM	0.6863	0.6745	0.6863	0.6317
	DL	0.6364	0.6321	0.6364	0.6340
	Hybrid-LGBM	0.6876	0.6844	0.6876	0.6248
	Hybrid-DL	0.6494	0.6391	0.6494	0.6429
60	LGBM	0.6876	0.6701	0.6876	0.6478
	DL	0.6391	0.6365	0.6391	0.6377
	Hybrid-LGBM	0.6958	0.7004	0.6958	0.6359
	Hybrid-DL	0.6329	0.6204	0.6329	0.6250
70	LGBM	0.6931	0.6858	0.6931	0.6414
	DL	0.6521	0.6472	0.6521	0.6494
	Hybrid-LGBM	0.6979	0.7060	0.6979	0.6374
	Hybrid-DL	0.6507	0.6426	0.6507	0.6458
80	LGBM	0.6876	0.6775	0.6876	0.6327
	DL	0.6391	0.6241	0.6391	0.6289
	Hybrid-LGBM	0.6910	0.6893	0.6910	0.6311
	Hybrid-DL	0.6651	0.6514	0.6651	0.6551
90	LGBM	0.6958	0.6894	0.6958	0.6462
	DL	0.6473	0.6413	0.6473	0.6439
	Hybrid-LGBM	0.6931	0.6912	0.6931	0.6357
	Hybrid-DL	0.6746	0.6624	0.6746	0.6656
100	LGBM	0.6972	0.6943	0.6972	0.6451
	DL	0.6644	0.6508	0.6644	0.6545
	Hybrid-LGBM	0.6924	0.6986	0.6924	0.6277
	Hybrid-DL	0.6801	0.6678	0.6801	0.6708

Table 1: Performance metrics of models across different training data percentages.

Figure 5 and Figure 6 compare the accuracy and F1 score of four models—LGBM, DL, Hybrid-LGBM, and Hybrid-DL across varying amounts of training data.

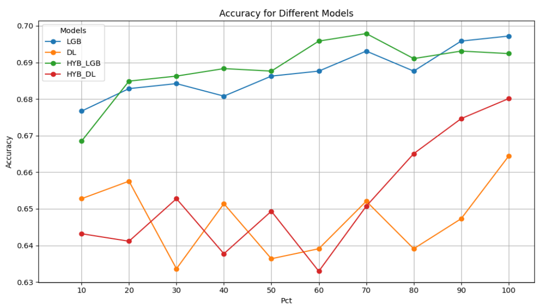

Figure 5: Accuracy

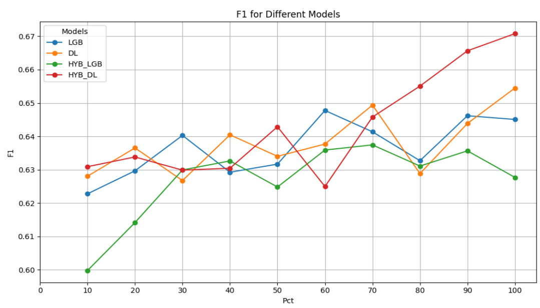

Figure 6: F1 Score

5 Conclusion

We have compared quantum–classical hybrid learning with classical learning architectures in datascarce environments. Evaluating LGBM, DL, Hybrid-LGBM, and Hybrid-DL, we observed that Hybrid-LGBM consistently delivers competitive performance while demonstrating superior data efficiency, achieving a maximum accuracy of 0.69 with 70% of training data. Future work may further improve accuracy by experimenting with alternative quantum circuits and varying the number of qubits.

6 Ethics

The dataset consists of publicly available, non-sensitive text corpora. Experiments comply with data licenses and research standards, with no human subjects involved, so ethical approval was not required. The hybrid quantum–classical framework is for research purposes only, and all references are acknowledged.

References

Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. 2017. Quantum machine learning. *Nature*, 549:195–202.

Harry Buhrman, Richard Cleve, John Watrous, and Ronald de Wolf. 2001. Quantum fingerprinting. *Physical Review Letters*, 87(16):167902.

S. M. Darwish, I. A. Mhaimeed, and A. A. Elzoghabi. 2023. A quantum genetic algorithm for building a semantic textual similarity estimation framework for plagiarism detection applications. *Entropy*, 25(9):1271.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. *Microsoft Research Technical Report (MRPC)*.

H. Gao, P. Zhang, J. Zhang, and C. Yang. 2024. A quantum-inspired hierarchical semantic interaction model for text classification. *Neurocomputing*, 611:128658.

R. Guarasci, G. De Pietro, and M. Esposito. 2022. Quantum natural language processing: challenges and opportunities. *Applied Sciences*, 12(11):5651.

Vojtěch Havlíček, Antonio D. Córcoles, Kristan Temme, Aram W. Harrow, Abhinav Kandala, Jerry M. Chow, and Jay M. Gambetta. 2019. Supervised learning with quantum-enhanced feature spaces. In *Nature*, volume 567, pages 209–212.

Ian T. Jolliffe. 2002. Principal component analysis (2nd ed.). In *Springer Series in Statistics*. Springer.

Guolin Ke, Qiwei Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qi Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Diederik P Kingma. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Vishu Madaan, Prateek Agrawal, Nandini Sethi, Vikas Kumar, and Sanjay Kumar Singh. 2016. A novel approach to paraphrase english sentences using natural language processing. *International Journal of Control Theory and Applications*, 9(11):5119–5128.

Konstantinos Meichanetzidis, Alexis Toumi, Giovanni de Felice, and Bob Coecke. 2023. Grammar-aware sentence classification on quantum computers. *Quantum Machine Intelligence*, 5(1):10.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.