Hybrid Quantum-Classical Machine Learning for Sentiment Analysis

Abu Kaisar Mohammad Masum, Anshul Maurya, Dhruthi Sridhar Murthy,
Pratibha, and Naveed Mahmud
Electrical Engineering and Computer Science
Florida Institute of Technology
{amasum2022, amaurya2022, dsridharmurt2022, pratibha2023}@my.fit.edu
{nmahmud}@fit.edu

Abstract—The collaboration between quantum computing and classical machine learning offers potential advantages in natural language processing, particularly in the sentiment analysis of human emotions and opinions expressed in large-scale datasets. In this work, we propose a methodology for sentiment analysis using hybrid quantum-classical machine learning algorithms. We investigate quantum kernel approaches and variational quantum circuit-based classifiers and integrate them with classical dimension reduction techniques such as PCA and Haar wavelet transform. The proposed methodology is evaluated using two distinct datasets, based on English and Bengali languages. Experimental results show that after dimensionality reduction of the data, performance of the quantum-based hybrid algorithms were consistent and better than classical methods.

Index Terms—Quantum Machine Learning, Haar Transform, SVM, Sentiment Analysis.

I. INTRODUCTION

Sentiment analysis or opinion mining is the process of analyzing digital text to determine the emotions and opinions embedded in the text, and helps shed light on important issues of human opinion [1]. By examining data from sources such as emails, tweets, chat transcripts, reviews, etc., sentiment analysis tools provide valuable insights into public opinions. It is a reliable method for predicting human behavior and has essential use cases such as improving customer service, brand monitoring, and product market research.

In recent years, researchers have demonstrated the efficiency of machine learning techniques for the correct classification of sentiments [2]. However, due to the computational limitations of classical machine learning [3], researchers are investigating the possibilities of newer, more efficient paradigms of computing, such as quantum computing. Quantum computers can take advantage of quantum mechanical effects such as superposition and entanglement to provide exponential speedup in specific tasks [4] compared to state-of-the-art classical computers. The rapid advancement of noisy intermediate-scale (NISQ) quantum hardware has driven research into quantum algorithm development for a variety of multidisciplinary fields such as Quantum Machine Learning (QML) [5]. Integrating machine learning with quantum computing can lead to greater benefits such as improved text classification [6]. Recent studies have introduced a quantum support vector machine (QSVM) for conventional data, to improve the performance of the traditional support vector machine (SVM) [7]. A QSVM uses the quantum kernel approach [8], which is essentially a hybrid approach that uses a classical SVM as the classifier. A type of quantum classifier known as the Variational Quantum Classifier (VQC) [9] has also been introduced that uses parameterized quantum circuits.

The limited scalability of current NISQ devices makes it challenging to encode high-dimensional data for QML tasks. Dimension reduction techniques can be used to decrease the data dimensionality by identifying the most pertinent features from data and/or demoting the input number of data points. Principal component analysis, also referred to as PCA, is a widely known approach for obtaining the principal features of the data [10]. Another data compression technique widely used in signal processing is the Wavelet Transform, which preserves the spatial locality of data, and compared to other transforms, also provides better computation speeds [11].

In this work, we investigate the use of quantum machine learning techniques for sentiment analysis of textual data, and we propose dimension reduction techniques to reduce the dimensionality of the textual data. We present a methodology combining conventional Natural Language Processing (NLP) techniques with QML models such as QSVM and VQC. The proposed methodology also integrates techniques such as PCA and Wavelet transform to reduce data dimensionality, making it convenient to fit the classical data onto the quantum models and reducing the algorithm execution time.

Our methodology is evaluated by employing textual data from two languages: English and Bengali. NLP for English text has been explored, for instance, Zhang et al. [12] investigated English text classification. The structure of the Bengali text is complex and different from other languages, especially in *pre-processing* and *word embedding* [13] phases. This work presents efficient models for *pre-processing* and *word embedding* in support of both Bengali and English language-based text. In addition, we investigate the efficiency of quantum models for feature mapping and classification, and compare their performance with classical methods, when applied for sentiment analysis. An important contribution of this research is the use of Wavelet transform for dimensionality reduction of textual data. Wavelet-based feature extraction enables a meaningful representation of the data with reduced dimensionality, and combined with hybrid quantum-classical ML models, leads to improved sentiment analysis.
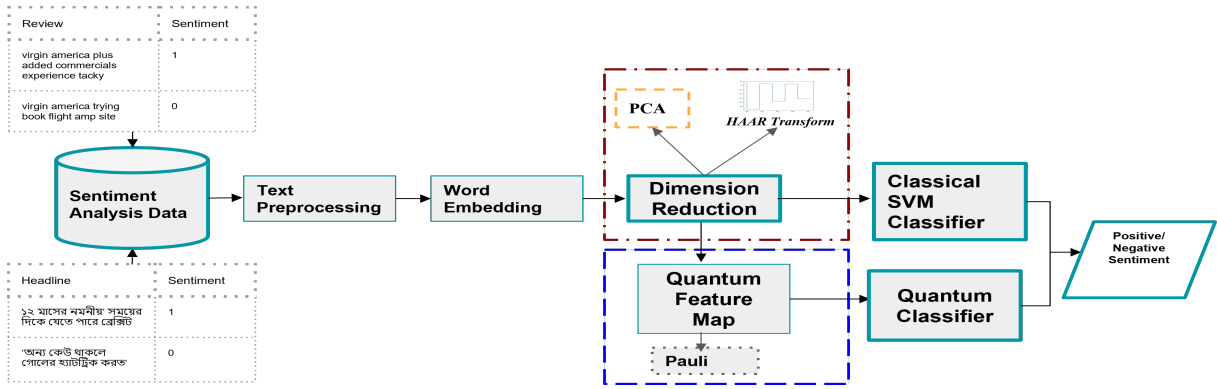
Fig. 1. Workflow for the proposed hybrid quantum-classical methodology for sentiment analysis.

II. RELATED WORK

In this section, we discuss the state-of-the-art methods (quantum or hybrid) that have been proposed for sentiment analysis. Ganguly et al. used the lambeq toolkit for sentiment analysis [14]. They performed noisy simulations on an IBM backend, achieving 83.33% accuracy. However, they used a very small dataset (130 sentences), and their test accuracy could be improved for noisy quantum simulation by performing more iterations. Morales et al. [15] proposed a framework for quantum natural language processing (QNLP) and implemented their work on near-term quantum computers. In their work, they leveraged variational quantum circuits in QNLP models. Further research and empirical studies are necessary to address the challenges involved and understand the potential benefits and limitations of applying quantum computing in natural language processing. In their paper, Meichanetzidis et al. [16] presented a pipeline for QNLP. The framework used is compositional distributional semantics (DisCoCat), which extends the compositional structure of group grammars. Overall, the paper provides an overview of the proposed QNLP pipeline, but it lacks empirical evidence, comparative analysis, and in-depth discussions on the feasibility and limitations of the approach.

Existing quantum sentiment analysis models primarily rely on rule-based techniques to process and analyze sentiment data. These models utilize quantum machine learning algorithms on a small amount of text data to extract sentiment. On the other hand, our hybrid quantum-classical sentiment analysis model follows automatic approaches for combining both quantum and classical computing techniques. Instead of solely relying on quantum algorithms, we leverage the strengths of classical algorithms alongside quantum algorithms to enhance sentiment analysis performance.

III. PROPOSED METHODOLOGY

In this work, we present a methodology for performing sentiment analysis of languages by combining classical and quantum machine learning techniques. At first, the input data undergoes pre-processing, then a word embedding algorithm converts pre-processed data to a word vector. After that, we apply dimension reduction algorithms to reduce data dimensionality. The consequent data is then fed to either classical SVM models, or to quantum models, for training and classification of sentiment. The workflow of the proposed methodology is shown in Fig. 1 and we discuss its further details in the rest of this section.

A. Sentiment Analysis Data

In this work, we have used two datasets for evaluating the methodology. The Bengali dataset [17] consists of news data where each line of text reflects human opinion. Scripts were used to collect the data from Bengali news portals. Furthermore, positive and negative sentiments have been separated into two categories of data. The dataset is structured as two columns of Bengali text and their correlated class and contains a total of 1619 data points and 65 distinct features. Each line of text contains vital words that represent the text context. Another dataset from *"Twitter US Airline Sentiment"* [18] has been used to measure the performance of the implemented model. The dataset has a total of 728 data points and 2027 unique features. For the Bengali and Twitter datasets, two separate word vector models subsequently converted them into two sets of feature vectors.

B. Text Pre-processing

Pre-processing Bengali text is distinct from other languages. Bengali words have a complex format. Therefore, additional study is needed to develop rule-based grammar systems for the Bengali language and we primarily focused on basic text preparation and automated systems.

We have implemented a function for pre-processing the Bengali text which includes techniques such as sentence-to-word [19], regular expression [20], and stop-word removal [21]. The text is processed from raw data using regular expression and then each sentence is split into words. At the same time, the stop words are removed from the list of stop words. Consequently, the processed text is sent to the next stage, i.e., word embedding, see Fig. 1. Fig. 2 gives a model diagram for the text pre-processing phase for Bengali text. For the Twitter dataset, we have applied available pre-processing approaches like tokenizer [19]. To facilitate the training of

data, the output column must be converted into binary labels. To achieve this, a label encoder based on scikit-learn is utilized to transform the output column into binary labels.
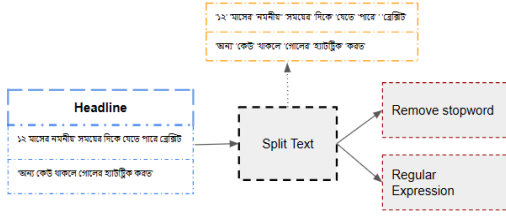

Fig. 2. Bengali text pre-processing method for sentiment analysis.

C. Word Embedding

We have leveraged the count vectorization technique [22] as a method of text data representation. By definition, count vectorization is a type of Boolean concept formulation in document labeling terminology, where documents can be represented via a set of words. In count vectorization, the dimension of the word vectors is determined by the length of the vocabulary. Whenever a new word occurs, the number of words and dimensions are simultaneously incremented by one. A counting vectorizer's fundamental objective is to comprehend and accommodate every single word in the lexicon. Fig. 3 shows a model of the count vectorizer used in our approach. A document phrase matrix has been constructed on the basis of the new lexicon that was developed. In our approach, every data row comprises a collection of words that signifies specific information. All the words within these data rows convey either negative or positive sentiments [23].
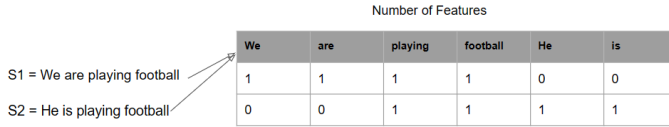

Fig. 3. Count vectorization approach for word embedding.

D. Dimension Reduction

For the proposed sentiment analysis methodology, we integrate two different techniques for data dimensionality reduction: Principal Component Analysis for dataset feature reduction and Haar Wavelet Transform for data compression.

a) Principal Component Analysis: In our PCA model, linear arrays are used at the beginning to determine the major vocabulary elements from the text datasets. The first principle component shows the initial set of data with the highest variance, while the second principle component shows the rest of the dataset with the most variation. The process continues sequentially, with the subsequent principal component representing the set of data from the rest of the dataset having the highest level of variance [24]. The most influential phrases are subsequently put to feature extraction procedures. M represents the overall count of principal components, while P denotes the substantial percentage of principal elements present in each principal component. However, P is commonly understood as the total count of principal elements present in the M-dimensional spaces. This represents the highest percentage of variable value pairs. The PCA principles provide clear evidence that $P \leq M$, as derived from the rules. This supports the data dimension reduction approach for our datasets. Nevertheless, PCA is the most useful data compression method for classical systems where the method depends on the corresponding vector dimension of the text.

b) Haar Wavelet Transformation: Haar transform holds significant prominence in the realms of signal and image processing [25]. Haar transform makes it possible to decrease the number of data points while preserving the locality of the data and the relationships between neighboring values. The motivation behind utilizing this compression technique in our approach is to reduce the dimensionality of the data, leading to faster quantum circuit execution time in the consequent stages of our proposed methodology. The inverse of the Haar transform [26] can also be used to get back the original data points after dimension reduction.

The Haar transform can be generalized as n-Dimensional (n-D) Haar transform, where n is the dimensionality of the data. Detailed algorithms for 1-D and 2-D Haar Transform can be found in [26]. In this work, we employed a 1-D discrete Haar Transform due to the discrete nature of the dataset and to reduce the dimensionality of the dataset. The 1-D Haar Transform can be performed as a multi-level, decomposable operation [26]. Each level of operation involves dividing the set of data points into two distinct non-overlapping segments. This division is achieved by computing the average and difference between neighboring values. The resulting difference values are generally close to zero or significantly smaller in comparison to the averaging values. Consequently, these negligible difference values can be safely discarded. By discarding them, the transformed dataset retains only the high-amplitude values, effectively reducing the dimension(s) of the original dataset by exactly half. For l decomposition levels, the dimension(s) will be reduced by 2^l.

E. Classical SVM Classifier

SVM is a classical machine learning method that allocates labels to data objects after training on example data [27]. For example, an SVM can be used for negative and positive sentiment analysis on a text-based dataset. The SVM method works by statistical classification where a separate hyperplane and a maximum number of margins between two separated objects work as base points. The linear SVM classifier is considered the most fundamental and quickest SVM, supposing a straight-line separation between classes. It was defined as the most effective technique for text categorization [28]. Our sentiment analysis methodology employs a traditional SVM classifier to accommodate each data point within n-dimensional spaces. Within these spaces, the feature values are extracted based on the corresponding data point's coordinates.

F. Quantum Feature Mapping

Quantum computers harness the power of an exponentially large quantum state space, offering great potential for han-

dling large datasets and mapping them to higher dimensions. Quantum feature mapping serves as a means to non-linearly map classical data onto quantum states, represented as vectors in Hilbert space, $|f(x)\rangle \langle f(x)|$; where f is a data mapping function and x is the feature data point [9].

A quantum circuit that is classically hard to simulate is important to gain the quantum advantage, as it provides a mapping of the data that is hard to compute classically. Entanglements between qubits in such circuits, take account of nonlinear interactions between features [9]. In our work, we have employed a generalized second-order Pauli feature mapping, discussed further:

a) Pauli feature map: A generalized method for feature mapping that utilizes Pauli basis gate set [4]. Based on the mapping function f, we design a unitary utilizing Pauli gate set {X, Y, and Z}, and an entanglement scheme which is typically either *linear* or *full* entanglement. The full entanglement feature map circuit will have full connectivity between qubits, accounting for $n(n-1)/2$ feature interactions, where n is the number of qubits in the circuit. Linear entanglement scheme accounts for interactions with only adjacent qubits. Thus, the full entanglement feature map has a higher circuit depth than the linear entanglement scheme. A classically hard to simulate [29] Pauli feature map unitary for the gate pair of Z and XX is shown in equation (1), accounting for two-qubit interaction.

$$U_{f(x)} = \left(\exp \left(i \sum_{q_j, q_k} f_{q_j, q_k}(x) X_{q_j} \otimes X_{q_k} \right) \times \exp \left(i \sum_{q_j} f_{q_j}(x) Z_{q_j} \right) H^{\otimes n} \right)^r \quad (1)$$

Here X and H are conventional NOT and Hadamard gates respectively, XX is the tensor of two X gates, and r is the number of circuit repetitions. The unitary $U_{f(x)}$ will be applied to $|0\rangle$, the initial state of the qubits. The H gate puts the circuit in the superposition state and phase gate (rotation gate in Pauli basis) manipulates the superposition state based on the chosen function map $f(x)$. In (1), the exponent involving the tensor operation $(X \otimes X)$ generates the entangled portion of the feature map circuit, and the remaining exponent terms, contribute to the rest of the non-entangled part of the circuit.

G. Quantum Classifiers

a) SVM classifier with quantum kernel: SVM can handle linearly separable data points but non-linearly separable data can be classified by employing kernel functions like linear, polynomial, and sigmoid functions, which maps the data to higher dimensions. The expression to calculate the quantum kernel matrix elements is $K_{i,j} = \mod (\langle f(x_i)|^{\dagger} |f(x_j)\rangle)^2$. $K_{i,j}$ is the measure of the distance between each data point with every other in the dataset. After computing the kernel matrix, it can be utilized as input for any standard classical machine-learning model to train a hyperplane.

b) Variational Quantum Classifier (VQC): A quantum algorithm for classification known as variational quantum classifier (VQC) was introduced in [9]. In the VQC algorithm, a variable parameterized quantum circuit, also known as ansatz, is used in conjunction with quantum feature maps for classification.

In VQC, the output of the quantum feature map is passed through the ansatz, which typically consists of parameterized rotation gates and CNOT gates [9]. Consequently, measurement of the circuit produces a classical bitstring that is mapped to a binary label, and then matched with the corresponding binary label of the encoded feature set in the feature map. Later this mapping gets utilized by a classical system which tunes the rotation parameters by optimizing the cost function $\langle f(x)| U^{\dagger} \hat{M} U |f(x)\rangle \geq \Delta$ [9], where U is the random unitary which initializes the parameters of ansatz and $\hat{M}$ is the measurement typically in a Z basis. The value of Δ decides the separation between the labels. In our work, we have used a real amplitude circuit as an ansatz with linear entanglement scheme. The real amplitude circuit is based on the Y-rotation gate (R_y) and only affects the real components of the quantum state.

IV. EXPERIMENT RESULTS AND ANALYSIS

The experiments to evaluate the proposed hybrid methodology were conducted using qiskit [30]. The experiments were performed on the noisy *qasm_simulator* [31] backend from IBM. For evaluation we used two distinct datasets containing English and Bengali text data. We classified text data into two classes, positive and negative based on the sentiment of the data.

To establish a baseline, we first performed sentiment analysis on the datasets using fully-classical SVM models without any dimension reduction. The obtained results are presented in Table I. For the Bengali dataset, the classical accuracy on Test data is 72%, while for the English dataset, the accuracy is 84%. In this work, we have used namely the following quantum-classical methods, the SVM classifier with quantum kernel and the Variational Quantum Classifier (VQC), which is a parameterized quantum circuit classifier. The VQC circuit parameters were optimized using ADAM optimizer [32]. For comparison, we used a third method called the Quantum Support Vector Classifier (QSVC), which is another variant of SVM classifier with a quantum kernel and is directly imported from qiskit's library. The QSVC is a pre-built qiskit function which takes a feature map input and trains an SVM classifier.

We implemented quantum circuit-based, classically-hard-to-simulate feature mapping by synthesizing (1). For our experiments, the data mapping function f used is:

$$f_{q_j, q_k}(x) = (\pi - x[j])(\pi - x[k]) \quad (2)$$

$$f_{q_m}(x) = x[m] \quad (3)$$

Equation (2), will be utilized for an entangled qubit pair, q_j and q_k, and x represents the feature data point, while (3) is applicable for non-entangled qubits. The entanglement arrangement we used in the experiments is linear entanglement.

In Table II, we present the performance of the implemented quantum-classical methods when using Pauli feature maps. We evaluate their effectiveness on the Bengali dataset reduced to 2, 4, and 6 feature sets after PCA. The test accuracy consistently falls within the range of 71% to 72% for all feature sets.

Dataset	Number of Data	Number of Features	Algorithm	Train (%) Accuracy	Test (%) Accuracy
Bengali News	1,619	65	SVM	72	72
Twitter US Airline	728	2,027	Classifier	99	84

TABLE I: Performance of Classical SVM for Bengali and Twitter Datasets.

Number of Qubits	Number of Data	SVM Classifier with Quantum Kernel			QSVC			VQC(100 iterations)		
		Test(%) Accuracy	Train(%) Accuracy	Time Elapsed in Training (sec)	Test(%) Accuracy	Train(%) Accuracy	Time Elapsed in Training (sec)	Test(%) Accuracy	Train(%) Accuracy	Time Elapsed in Training (sec)
2		71	69	3,455.40	72	69	4,456	72	68	618.6
4	1619	71	69	6,613.60	72	69	7,216	72	68	1866.6
6		71	69	8,007	72	69	8,679	72	68	3,984

TABLE II: Performance comparison of quantum-classical methods using Pauli feature map and PCA on Bengali Dataset.

Number of Qubits	Number of Data	SVM Classifier with Quantum Kernel			QSVC			VQC(100 iterations)		
		Test(%) Accuracy	Train(%) Accuracy	Time Elapsed in Training (sec)	Test(%) Accuracy	Train(%) Accuracy	Time Elapsed in Training (sec)	Test(%) Accuracy	Train(%) Accuracy	Time Elapsed in Training (sec)
2		70	69	384	71	70	676.2	71	70	249
4	728	71	69	840	71	70	1150.8	71	70	747
6		70	69	2,071.20	71	70	2,071.20	71	70	1,584

TABLE III: Performance comparison of quantum-classical methods using Pauli feature map and PCA on Twitter Dataset.

Number of Decomposition Levels (HAAR Transform)	Number of Training Data	SVM Classifier with Quantum Kernel			QSVC			VQC (100 iterations)			Classical Accuracy	
		Test (%) Accuracy	Train (%) Accuracy	Time Elapsed in Training (sec)	Test (%) Accuracy	Train (%) Accuracy	Time Elapsed in Training (sec)	Test (%) Accuracy	Train (%) Accuracy	Time Elapsed in Training (sec)	Test (%) Accuracy	Train (%) Accuracy
0	582	71.23	70.44	1270.88	71.23	70.44	212.36	71.23	71.23	303.09	84.93	99.65
1	291	71.23	67.69	403.84	71.23	67.69	100.91	71.23	67.70	155.27	58.21	100
2	145	71.23	68.27	165.42	71.23	68.27	51.75	71.23	68.28	88.21	67.12	100
3	72	71.23	66.66	62.55	71.23	83.11	12.15	71.23	66.67	46.56	36.37	100
4	35	71.23	62.85	28.48	71.23	62.85	3.43	71.23	62.86	23.63	54.79	100
5	17	71.23	64.7	13.58	71.23	64.7	1.2	71.23	64.71	12.8	28.76	100

TABLE IV: Performance comparison using Pauli feature Map and PCA + Haar compression on Twitter Dataset.

Number of Decomposition Levels (HAAR Transform)	Number of Training Data	SVM Classifier with Quantum Kernel			QSVC			VQC (100 iterations)			Classical Accuracy	
		Test (%) Accuracy	Train (%) Accuracy	Time Elapsed in Training (sec)	Test (%) Accuracy	Train (%) Accuracy	Time Elapsed in Training (sec)	Test (%) Accuracy	Train (%) Accuracy	Time Elapsed in Training (sec)	Test (%) Accuracy	Train (%) Accuracy
0	1295	72.22	67.56	2534.04	72.22	67.56	2266.49	72.22	67.57	375.32	72.22	67.56
1	647	72.22	65.53	1178.75	72.22	65.53	1232.35	72.22	65.22	367.6	72.22	65.53
2	323	72.22	65.94	136.7	72.22	65.94	136.39	72.22	65.94	95.838	72.22	65.94
3	161	72.22	68.32	33.41	72.22	68.94	33.1	72.22	68.94	47.96	72.22	71.42
4	80	72.22	73.75	8.503	72.22	73.75	8.06	72.22	71.25	23.8	72.22	80
5	40	72.22	65	2.24	72.22	65	2.04	72.22	65	11.32	72.22	87.5

TABLE V: Performance comparison using Pauli feature Map and PCA + Haar compression on Bengali Dataset.

We also measured the training times for each technique, and VQC required the least training time for each feature set compared to other algorithms, see Table II. The results in Table III for the Twitter dataset using Pauli feature maps, respectively demonstrate consistent accuracy for both training (69%-71%) and testing (70%-71%). Notably, when training the feature sets, the QSVC and VQC algorithms performed with higher test and train accuracies compared to the SVM classifier with quantum kernel, see Table III. Furthermore, VQC exhibits shorter training times compared to other algorithms when applied to this dataset also.

With the number of features reduced to two using PCA, we applied Haar transform to further compress the number of data points in the datasets. This significantly reduced the training time of the subsequent quantum methods. Up to five levels of Haar decomposition were performed. In addition to the quantum-classical methods, we also evaluate a classical SVM classifer for accuracy comparison with the quantum-classical methods. Table IV show that only one level of decomposition reduced the classical SVM test accuracy to about 58% with further degradation at higher decomposition levels. On the other hand, the quantum-classical algorithms provide better and steady classification accuracy of 71.23%. This implies that the Haar transform in combination with the quantum-classical methods is more effective than when applied for the classical SVM. The VQC algorithm takes less time to train than other methods on average and maintains consistent classification accuracy up to the maximum number of decomposition levels applied. For the Bengali dataset, see Table V, the classical accuracy is constant for up to the maximum decomposition level because the dataset used in the experiment is balanced. Quantum-classical algorithms for the Bengali dataset after

dimensionality reduction show the same level of performance as classical systems without dimension reduction, achieving an accuracy of 72.22%, see Tables I, and V. This implies that for the Bengali dataset, our quantum-classical methods in combination with dimension reduction was highly effective, leading to efficient sentiment analysis on a large dataset.

V. Conclusion and Future Work

Our research demonstrates the potential of quantum-classical hybrid algorithms in sentiment analysis of Bengali and English datasets. We also highlighted the importance of considering language-specific characteristics, particularly in Bengali sentiment analysis. We investigated hybrid quantum-classical algorithms that included quantum feature maps and quantum classifiers. Moreover, we integrated dimension reduction techniques to facilitate encoding classical data on to the quantum models. Moving forward, there are several areas for future work in quantum-classical hybrid sentiment analysis research. Firstly, expanding the analysis to include other languages would provide a more comprehensive understanding of the effectiveness of hybrid techniques across different linguistic contexts. Additionally, exploring error mitigation and noise correction, and the translation of quantum sentiment analysis into practical quantum applications are important areas for further investigation.

References

[1] Bing Liu. *Sentiment analysis and opinion mining*. Springer Nature, 2022.

[2] Hamed Taherdoost and Mitra Madanchian. Artificial intelligence and sentiment analysis: A review in competitive research. *Computers*, 12(2):37, 2023.

[3] Hsin-Yuan Huang, Richard Kueng, and John Preskill. Information-theoretic bounds on quantum advantage in machine learning. *Physical Review Letters*, 126(19):190505, 2021.

[4] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[5] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, 2017.

[6] Syed Mustajar Ahmad Shah, Hongwei Ge, Sami Ahmed Haider, Muhammad Irshad, Sohail M Noman, Jehangir Arshad Meo, Asfandeyar Ahmad, and Talha Younas. A quantum spatial graph convolutional network for text classification. *Comput. Syst. Sci. Eng.*, 36(2):369–382, 2021.

[7] Rui Zhang, Jian Wang, Nan Jiang, and Zichen Wang. Quantum support vector machine without iteration. *Information Sciences*, 635:25–41, 2023.

[8] Jae-Eun Park, Brian Quanz, Steve Wood, Heather Higgins, and Ray Harishankar. Practical application improvement to quantum svm: theory to practice. *arXiv preprint arXiv:2012.07725*, 2020.

[9] Vojtěch Havlíček, Antonio D Córcoles, Kristan Temme, Aram W Harrow, Abhinav Kandala, Jerry M Chow, and Jay M Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212, 2019.

[10] LJ Cao, Kok Seng Chua, WK Chong, HP Lee, and QM Gu. A comparison of pca, kpca and ica for dimensionality reduction in support vector machine. *Neurocomputing*, 55(1-2):321–336, 2003.

[11] Esam El-Araby, Tarek El-Ghazawi, Jacqueline Le Moigne, and Kris Gaj. Wavelet spectral dimension reduction of hyperspectral imagery on a reconfigurable computer. In *Proceedings. 2004 IEEE International Conference on Field-Programmable Technology (IEEE Cat. No. 04EX921)*, pages 399–402. IEEE, 2004.

[12] Wen Zhang, Taketoshi Yoshida, and Xijin Tang. A comparative study of tf* idf, lsi and multi-words for text classification. *Expert systems with applications*, 38(3):2758–2765, 2011.

[13] Sheikh Abujar, Mahmudul Hasan, MSI Shahin, and Syed Akhter Hossain. A heuristic approach of text summarization for bengali documentation. In *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–8. IEEE, 2017.

[14] Srinjoy Ganguly, Sai Nandan Morapakula, and Luis Miguel Pozo Coronado. Quantum natural language processing based sentiment analysis using lambeq toolkit. In *2022 Second International Conference on Power, Control and Computing Technologies (ICPC2T)*, pages 1–6. IEEE, 2022.

[15] John A Morales, E Orduña, C Rehtanz, RJ Cabral, and AS Bretas. Comparison between principal component analysis and wavelet transform 'filtering methods for lightning stroke classification on transmission lines. *Electric power systems research*, 118:37–46, 2015.

[16] Konstantinos Meichanetzidis, Stefano Gogioso, Giovanni De Felice, Nicolò Chiappori, Alexis Toumi, and Bob Coecke. Quantum natural language processing on near-term quantum computers. *arXiv preprint arXiv:2005.04147*, 2020.

[17] Bengalidata. https://www.kaggle.com/datasets/kaisermasum24/bengali-news-headline-sentiment. Accessed: July 2023.

[18] Prayag Tiwari, Pranay Yadav, Sachin Kumar, Brojo Kishore Mishra, Gia Nhu Nguyen, Sarada Prasad Gochhayat, Jagendra Singhk, and Mukesh Prasad. Sentiment analysis for airlines services based on twitter dataset. *Social Network Analytics: Computational Research Methods and Techniques*, 149, 2018.

[19] Gregory Grefenstette. Tokenization. In *Syntactic wordclass tagging*, pages 117–133. Springer, 1999.

[20] Menglin Cui, Ruibin Bai, Zheng Lu, Xiang Li, Uwe Aickelin, and Peiming Ge. Regular expression based medical text classification using constructive heuristic approach. *IEEE Access*, 7:147892–147904, 2019.

[21] Catarina Silva and Bernardete Ribeiro. The importance of stop word removal on recall values in text categorization. In *Proceedings of the International Joint Conference on Neural Networks, 2003.*, volume 3, pages 1661–1666. IEEE, 2003.

[22] Ye Wang, Zhi Zhou, Shan Jin, Debin Liu, and Mi Lu. Comparisons and selections of features and classifiers for short text classification. In *Iop conference series: Materials science and engineering*, volume 261, page 012018. IOP Publishing, 2017.

[23] Turki Turki and Sanjiban Sekhar Roy. Novel hate speech detection using word cloud visualization and ensemble learning coupled with count vectorizer. *Applied Sciences*, 12(13):6611, 2022.

[24] Harun Uğuz. A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. *Knowledge-Based Systems*, 24(7):1024–1032, 2011.

[25] Wanli Ouyang, Renqi Zhang, and Wai-Kuen Cham. Fast pattern matching using orthogonal haar transform. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3050–3057. IEEE, 2010.

[26] Naveed Mahmud and Esam El-Araby. Dimension reduction using quantum wavelet transform on a high-performance reconfigurable computer. *International Journal of Reconfigurable Computing*, 2019:1–14, 2019.

[27] William S Noble. What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567, 2006.

[28] Rui Xia, Chengqing Zong, and Shoushan Li. Ensemble of feature sets and classification algorithms for sentiment classification. *Information sciences*, 181(6):1138–1152, 2011.

[29] Leslie Ann Goldberg and Heng Guo. The complexity of approximating complex-valued ising and tutte partition functions. *computational complexity*, 26:765–833, 2017.

[30] qiskit. https://qiskit.org/. Accessed: July 2023.

[31] Qiskit simulator. https://qiskit.org/ecosystem/aer/stubs/qiskit_aer.QasmSimulator.html. Accessed: July 2023.

[32] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.