Bi-Quantum Long Short-Term Memory for Part-of-Speech Tagging

Shyambabu Pandey
National Institute of Technology Silchar
shyambabu21_rs@cse.nits.ac.in

Partha Pakray
National Institute of Technology Silchar
partha@cse.nits.ac.in

Abstract

Natural language processing (NLP) is a subfield of artificial intelligence that enables computer systems to understand and generate human language. NLP tasks involved machine learning and deep learning methods for processing the data. Traditional applications utilize massive datasets and resources to perform NLP applications, which is challenging for classical systems. On the other hand, Quantum computing has emerged as a promising technology with the potential to address certain computational problems more efficiently than classical computing in specific domains. In recent years, researchers have started exploring the application of quantum computing techniques to NLP tasks. In this paper, we propose a quantum-based deep learning model, Bi-Quantum long short-term memory (BiQLSTM). We apply POS tagging using the proposed model on social media code-mixed datasets.

1 Introduction

Natural language processing (NLP) (Jurafsky, 2000) is a promising artificial intelligence (Russell and J, 2010) area that focuses on the interaction between human languages and computer systems. Its primary aim is to enable computer systems to understand, process, and generate human languages meaningfully. The extensive use of NLP applications makes NLP ubiquitous. From virtual assistants and chatbots to language translation and sentiment analysis, NLP consistently enhances communication, decision-making, and convenience of everyday interactions. NLP tasks involve computational linguistic and machine learning models to process and analyze text information.

Machine learning (Janiesch et al., 2021) is a subfield of artificial intelligence that focuses on developing algorithms and models to learn the relation between data and patterns from a dataset. It enables computer systems to make decisions, predictions, and recommendations by recognizing patterns and relationships between data. Neural network (Sharkawy, 2020) is a part of machine learning algorithms whose computational methodology is inspired by the working principle of the human brain. It is composed by combining a linear sequence of neurons.

In the last decade, machine learning has significantly appeared in advanced learning algorithms, which show remarkable data processing performance. The performance of machine learning algorithms is tremendous in various domains, especially in the NLP domain. The higher performance of NLP applications needs vast datasets and complex algorithms for training. However, the increasing data size requires extensive resources for training the traditional algorithms, leading to massive time and resource consumption.

On the other hand, Quantum computers can solve these challenges classical systems face. Due to their unique characteristics, superposition and entanglement can speed up computational time and decrease the demand for extensive resources. We can take leverage of quantum computing speedup by combining it with classical machine learning, known as quantum machine learning (QML) (Maria Schuld, 2021).

The main goal of QML algorithms is to apply quantum computing's distinct attributes to build machine learning algorithms that can surpass their classical counterparts in capability and efficiency. QML offers various superiority, including speedup runtime, increased learning efficiency by employing fewer data for training and finding complex patterns, and enhanced learning capability by improving the content-addressable memory (Phillipson, 2020). These advancements in QML make it applicable to domains like NLP.

In this paper, we proposed a quantum-inspired deep neural network model bidirectional quantum long short-term memory (BiQLSTM). We also apply the proposed system for NLP application, part-of-speech (POS) tagging on code-mixed datasets. POS tagging is a fundamental task that automatically assigns parts of speech in given sentences. POS tagging helps to analyze the syntactic and semantic structure of sentences. Most POS tagging techniques apply machine learning algorithms that automatically predict and assign the actual parts of speech of words according to their context within a sentence.

The paper's structure is as follows: section 2 explains the background of quantum computing and QML. Section 3 presents the related work about QML algorithms for NLP applications. The proposed QRNN model is presented in section 4. The description of the dataset is given in section 5. Section 6 shows the experiment and results. This paper concludes with a conclusion in section 7.

2 Background

2.1 Quantum computing

Quantum computing (Gyongyosi and Imre, 2019) is a branch of computer science that uses the principles of quantum theory, such as superposition and entanglement, to process information. It can solve complex problems, which are interactable for classical systems. Quantum algorithms show the power of quantum speedup, such as Grover's algorithms (Mandviwalla et al., 2018), to search an element from the unsorted database, which shows quadratic speedup compared to its classical solutions.

Quantum bit (qubit) is an essential information processing unit for quantum computing analogous to classical bit. The computational basic state of a qubit is $|0\rangle = \begin{bmatrix}1\\0\end{bmatrix}$ and $|1\rangle = \begin{bmatrix}0\\1\end{bmatrix}$. A single qubit can be in combination of basic states $|0\rangle$ and $|1\rangle$, such as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix}\alpha\\\beta\end{bmatrix}$, where α and β are complex numbers, known as amplitudes of quantum states. Any valid qubit must hold this condition $|\alpha|^2 + |\beta|^2 = 1$.

Quantum systems can exist in multiple-qubits states. The tensor product ($\otimes$) is applied to represent multiple-qubit states from several independent single-qubit states. For example, two individual quantum states $|\psi\rangle = \begin{bmatrix}\alpha_0\\\beta_0\end{bmatrix}$ and $|\phi\rangle = \begin{bmatrix}\alpha_1\\\beta_1\end{bmatrix}$, the tensor product of quantum states can be described as their collective states.

$$|\psi\phi\rangle = |\psi\rangle\otimes|\phi\rangle = \alpha_{00}|00\rangle+\alpha_{01}|01\rangle+\alpha_{10}|10\rangle+\alpha_{11}|11\rangle \tag{1}$$

Here two-qubit system generate four (2^2) basic states ($|00\rangle, |01\rangle, |10\rangle, |11\rangle$), just like that n-qubit quantum system can generate 2^n basic states ($|0\rangle$.......$|2^n - 1\rangle$).

$$|\psi\rangle = \sum_{j=0}^{2^n-1} \alpha_j|j\rangle \qquad and \qquad \sum_{j=0}^{2^n-1} |\alpha_j|^2 = 1 \tag{2}$$

Superposition is a fundamental property of quantum theory, enabling quantum systems to be simultaneously present in all possible qubit states. For example a single qubit can be in $|0\rangle$ and $|1\rangle$ at the same time ($|\psi\rangle = a|0\rangle + b|1\rangle$). Superposition in quantum mechanics enables a quantum system to exist simultaneously in multiple states. For instance, if an n-qubit quantum system encompasses 2^n computational states, a quantum system can store all these computational states concurrently. In contrast, classical systems can only hold a single value at any given time. The main benefit of superposition is it enables quantum computing to perform parallelism and increase exponential computational power.

Entanglement is another fundamental characteristic of quantum theory, which interconnects two or more quantum particles such that the state of one particle is dependent on each other. In quantum computing, changes in the state of one entangled qubit can immediately change the state of another. So, processing one entangled qubit can reveal the information of other paired qubits. Therefore, entanglement can enhance the processing speed of quantum computers. Entanglement is essential for quantum algorithms, which can offer speedup compared to their classical counterpart.

For any computing methodology, operators are required to process the information. In the case of quantum computing, quantum gates are fundamental operators for manipulating qubits. A Quantum gate is the basic building block of a quantum circuit, which is used for performing specific tasks. It is analogous to the classical gate, which transforms qubits' states linearly. The behavior of quantum gates is described by unitary matrices ($U^\dagger = U^{-1}$), which produce other valid qubit states.

$$U|\psi\rangle = |\phi\rangle \tag{3}$$

2.2 Quantum machine learning

QML algorithms are implemented by parameterized quantum circuits, which are constructed by quantum gates. Parameterized quantum circuit combines single and multiple qubit quantum gates. Each QML algorithm is divided into parts, encoding classical information into quantum states, processing the encoded information by variational quantum circuits, and measuring the outcome, which generates data labels.

Quantum computers operate only on quantum data, so we must encode classical data into quantum states. This encoding process, known as encoding methods, plays a crucial role in designing QML algorithms. It maps classical information into the Hilbert space of quantum systems, making it compatible with quantum computing. The selection of an encoding method significantly affects the performance of QML algorithms.

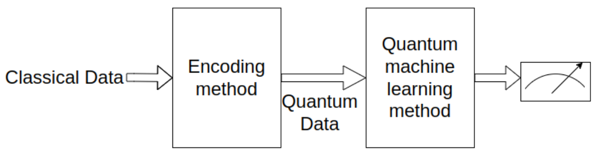

Figure 1: Overview of Quantum machine learning

Once the dataset features are encoded to a quantum state, quantum data is subjected to further process. An important component of QML algorithms, the variational quantum circuit (Benedetti et al., 2019) (VQC), is used to represent QML models. VQC comprises a series of quantum gates that are parameterized by tunable parameters. The main goal of using VQC is used to find the optimal values of variables that minimize the cost function. The cost function can represent a wide range of optimization problems, such as finding the lowest energy state of a molecular system or optimizing the parameters of a machine learning model.

VQC can be trained using a technique called quantum gradient descent, which is similar to classical gradient descent but uses quantum circuits to compute the cost function gradients for the variables (Mitarai et al., 2018). The gradients are then used to update the values of the variables in each iteration of the optimization process. One of the advantages of VQC is its ability to use quantum computing to solve optimization problems that are intractable for classical computers.

3 Related work

In this section, we discuss the related work for QML algorithms and their applications in the NLP field. QML algorithms include quantum-inspired traditional machine learning and deep learning models. The primary goal of using QML is to enhance the performance of classical machine learning algorithms.

Riordan et al. (O'Riordan et al., 2020) proposed a hybrid quantum-classical model, which can encode, interpret, and decode the meaning of sentences within the quantum circuit. The authors have shown how one can interpret the meanings of sentences and find similarities among various sentences' meanings. However, this model has limitations, such as when the data size is increased, the depth of the circuit is increased. So, the proposed model is well-suitable for the small and straightforward corpus.

Shi et al. (Shi et al., 2023) introduced a quantum-enhanced neural network for text classification for binary labels. The authors applied quantum-inspired complex-valued word embedding (ICWE). Generally, words have different meanings and equivalents, similar to a quantum system, which can hold several values. The authors have shown the effectiveness and feasibility of the applicability of the proposed methods for NLP tasks and can provide a solution for text information loss.

Quantum neural network is another research area of QML that may be applicable for NLP tasks such as POS tagging. Sipio et al. (Di Sipio et al., 2022) introduce quantum-based long short-term memory (QLSTM) for NLP applications such as POS tagging. This model is a demonstrated model, which shows that quantum-based neural network models apply to NLP tasks. They have performed POS tagging only on two sentences. They offer a comparison between QLSTM and classical LSTM results. This model has some limitations because it can not apply to large datasets.

Pandey et al. (Pandey et al., 2022) used the QLSTM (Di Sipio et al., 2022) model to perform POS tagging on the Mizo dataset, where Mizo is a low-resource language of India. The author experimented on a Mizo dataset with 47 POS tags and 30000 words. However, the result of their experiment could be better. The authors have tried with different numbers of qubits and other local simulators. They have investigated that current quantum devices do not apply to large datasets.

Pandey et al. (Pandey et al., 2023) introduced a hybrid quantum-classical QLSTM model. The proposed model is applied for POS tagging on social media code-mixed languages. Their experiment result shows the accuracy of QLSTM is better than classical LSTM. They have performed nine experiments on nine different datasets, which contain various Indian languages mixed with English languages.

Quantum natural language processing (QNLP) (Coecke et al., 2020) is another framework that utilizes properties of quantum mechanics for NLP applications. QNLP focuses on categorical quantum mechanics that represent the syntactic structure of sentences into states of quantum systems and grammar into entanglement. Lambeq (Kartsaklis et al., 2021) is an open-source software platform of QNLP, which is used to process several NLP applications.

4 BiQLSTM

Recurrent neural network (RNN) (Hibat-Allah et al., 2020) is a neural network variant for handling sequential data. However, the performance of the RNN model is not satisfactory for prolonged dependencies in text data. The RNN model suffers from a problem known as a vanishing gradient problem when data dependencies increase. Therefore, another RNN variant is helpful to handle prolonged dependencies and long short-term memory (LSTM) (Lobo Neto et al., 2020). LSTM is well known for managing longer dependencies in text data. It can resolve the vanishing gradient problem and is widely used in NLP applications.

LSTM utilizes memory cells for storing extended-period information. The working principle of the memory cell depends on gate units, input gate, forget gate, output gate, and update gate. The work of the input gate is to recognize new information, which will be accepted by the memory cell, and the forget gate selects information that will be discarded. After receiving wanted data and rejecting unwanted data, the update gate updates the new information. At the last, the output gate controls information stored in the memory cell, which will generate the output. Working these gates enables LSTM to select helpful information and forget useless data to handle prolonged dependencies in the dataset. The below equations show the flow of information in LSTM.

$$i_t = \sigma(W_i[h_{t-1}, X_t] + b_i) \tag{4}$$

$$f_t = \sigma(W_f[h_{t-1}, X_t] + b_f) \tag{5}$$

$$o_t = \sigma(W_o[h_{t-1}, X_t] + b_o) \tag{6}$$

$$S_t = tanh(W_s[h_{t-1}, X_t] + b_c) \tag{7}$$

$$h_t = o_t * tanh(S_t) \tag{8}$$

Where σ is the sigmoid function, i_t represents the workflow of the input gate, f_t represents the workflow of the forget gate, o_t represents the workflow of the output gate, and S_t represents the workflow of the update gate. W_i, W_f, W_o, and W_s denotes the parameters of corresponding gate. $[h_{t-1}, X_t]$ is combination of hidden state and input data. b_n is corresponding bias of W_n, where n is type of gate.

LSTM is built by stacking hidden cells and memory cells. Handling code-mixed language is complicated because mixed language contains different syntax. Therefore, Bidirectional Long Short-Term Memory (BiLSTM) networks are often considered more powerful and versatile than traditional LSTM networks due to their ability to capture contextual information in both forward and backward directions.

We propose a quantum-based BiLSTM (BiQLSTM) to implement NLP applications. The proposed model is an extended version of QLSTM (Pandey et al., 2023). Here, we convert the classical layer of BiLSTM into quantum circuits to enhance the performance. After converting each layer of BiLSTM into the quantum circuit, we stack each layer to build the proposed systems.

In this model, stochastic gradient descent (Sweke et al., 2020) is used as an optimizer that is best suitable for large datasets, and cross-entropy is used as a loss function. POS tagging is a multilabel classification (Vetagiri et al., 2023), so we use cross-entry as a function.

5 Database description

In our experiment, we use social media code-mixed datasets (Jamatia et al., 2015). Code-mixed languages are typically involved in mixing multiple languages and are commonly used in multivariate societies such as India. Processing such languages

Algorithm 1 Propsed algorithm for BiQLSTM

Input: $\mathbf{x\_t}$ - input at time t
Input: $\mathbf{h_{t-1}}$ - previous cell output at time $t-1$
Input: $\mathbf{c}_{t-1}$ - previous cell state at time $t-1$
Input: $\mathbf{W}_f, \mathbf{W}_i, \mathbf{W}_o, \mathbf{W}_c$ - weight matrices for forget, input, output, and cell gates
procedure ENCODE(X)
  for i to n_qubit **do**
    $Hadamard(i)$
    $RY(x\_i)$
  end for
end procedure
procedure VQC(X, α)
  for i to n_qubit **do**
    $CNOT(i, i+1)$
    $RY(\alpha\_i)$
  end for
end procedure
procedure BIQLSTM(INPUT,N_QUBIT, EMBEDDING_DIM,HIDDEN_LAYER)
  $\mathbf{f}_t = \sigma(VQC(\mathbf{W\_f}, Encode(\mathbf{x}_t, h_{t-1}))$
  $\mathbf{i}_t = \sigma(VQC(\mathbf{W\_i}, Encode(\mathbf{x}_t, h_{t-1}))$
  $\mathbf{o}_t = \sigma(VQC(\mathbf{W\_o}, Encode(\mathbf{x}_t, h_{t-1}))$
  $\mathbf{c}_t = \tanh(VQC(\mathbf{W\_c}, Encode(\mathbf{x}_t, h_{t-1}))$
  $\mathbf{c}_t = \mathbf{f_t} \odot \mathbf{c_{t-1}} + \mathbf{i_t} \odot \tilde{\mathbf{c_t}}$
  $\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t)$
  Output: $\mathbf{h}_t$ -
  Output: $\mathbf{c}_t$ state at time t
  Forward LSTM : $(\mathbf{h}_f^t, \mathbf{c}_f^t) \leftarrow \mathbf{x}_t,\ \mathbf{h}_f^{t-1},\ \mathbf{c}_f^{t-1}$
  Backward LSTM: $(\mathbf{h}_b^{T-t}, \mathbf{c}_b^{T-t}) \leftarrow \mathbf{x}_t,\ \mathbf{h}_b^{T-t+1}, \mathbf{c}_b^{T-t+1}$
end procedure

Table 1: POS Tagset

Category	Type	Description
Noun(G_N)	N_NN	Common Noun
	N_NNV	Verbal Noun
	N_NST	Spatio-temporal
	N_NNP	Proper Noun
Verb(G_V)	V_VM	Main
	V_VAUX	Auxiliary
Pronoun(G_PRP)	PR_PRP	Personal
	PR_PRL	Relative
	PR_PRF	Reflexive
	PR_PRC	Reciprocal
	PR_PRQ	Wh-Word
Adjective (G_J)	JJ	Adjective
Adverb (G_R))	RB_ALC	Locative Adverb
	RB_AMN	Adverb of Manner
Demonstrative(G$_PRP$)	DM_DMD	Absolute
	DM_DMI	Indefinite
	DM_DMQ	Wh-word
	DM_DMR	Relative
Quantifier(G_SYM)	QT_QTF	General
	QT_QTC	Cardinal
	QT_QTO	Ordinal
Particles(G_PRT)	RP_RPD	Default
	RP_NEG	Negation
	RP_INTF	Intensifier
	RP_INJ	Interjection
Residual (G$_X$)	RD_RDF	Foreign Word
	RD_SYM	Symbol
	RD_PUNC	Punctuation
	RD_UNK	Unknown
	RD_ECH	Echo Word
Conjunction	CC	Conjunction
Postposition	PSP	Pre-/Postposition
Determiner	DT	Determiner

is very difficult and challenging because the same text contains more than one language.

The dataset was collected from social media platforms like Twitter, Facebook, and WhatsApp. This code-mixed consists of a combination of Hindi and English language. It is further divided into two types according to the kinds of scripts, namely Devanagari and Romanized.

Table 1 shows the tagset of the dataset. Tagset is divided into two categories: Fine-grained and course-grained. The first column of the table shows a course-grained tagset, and the second column shows a fine-grained dataset. In our experiment, we use a fine-grained dataset.

Table 2: Accuracy of Experiment

Database	Code-mixed	Accuracy
Romanized dataset	Hindi-English	55.78
Devnagari dataset	Hindi-English	48.69

6 Experiment & Result

We develop a quantum-based BiLSTM for NLP applications. The proposed model is applied for POS tagging on social media code-mixed. According to their script type, we have performed two experiments on two different datasets, Devanagari and Romanized.

In our experiments, we use the Pennylane platform, a local simulator. Pennylane is an open-source software framework for Quantum computing and QML. It is developed for work and experiments with other Quantum computing platforms.

We use four qubits in this experiment. We chose four qubits because we want fewer qubits possible for NLP applications in quantum experiments. The number of epochs is 300. We observe that for more than 300 epochs, the model is overfitted.

The accuracy of both experiments is given in Table 2. We observe that the accuracy of the investigation of the Romanized dataset is higher than that of the Devnagari script datasets. The proposed system is more supportable for Romanized script. So, we will develop this model for other scripts, too.

7 Conclusion

NLP is a promising area of artificial intelligence that makes computer systems understand human languages. NLP applications involve machine learning and deep learning models to process text data. However, the recent machine learning model consumes more resources and demands a vast dataset. On the other hand, QML enhances the performance of NLP tasks. In this paper, we proposed a quantum-based BiQLSTM to perform pos tagging on code-mixed languages. This proposed model shows that a quantum-based deep learning model can handle NLP applications.

Acknowledgements

The work presented here falls under the Research Project Grant Ref. No. N-21/17/2020-NeGD supported by MeitY Quantum Computing Applications Lab (QCAL).

References

Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. 2019. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4):043001.

Bob Coecke, Giovanni de Felice, Konstantinos Meichanetzidis, and Alexis Toumi. 2020. Foundations for near-term quantum natural language processing. *arXiv preprint arXiv:2012.03755*.

Riccardo Di Sipio, Jia-Hong Huang, Samuel Yen-Chi Chen, Stefano Mangini, and Marcel Worring. 2022. The dawn of quantum natural language processing. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8612–8616.

Laszlo Gyongyosi and Sandor Imre. 2019. A survey on quantum computing technology. *Computer Science Review*, 31:51–71.

Mohamed Hibat-Allah, Martin Ganahl, Lauren E Hayward, Roger G Melko, and Juan Carrasquilla. 2020. Recurrent neural network wave functions. *Physical Review Research*, 2(2):023358.

Anupam Jamatia, Björn Gambäck, and Amitava Das. 2015. Part-of-speech tagging for code-mixed english-hindi twitter and facebook chat messages. In *Proceedings of the international conference recent advances in natural language processing*, pages 239–248.

Christian Janiesch, Patrick Zschech, and Kai Heinrich. 2021. Machine learning and deep learning. *Electronic Markets*, 31(3):685–695.

Dan Jurafsky. 2000. *Speech & language processing*. Pearson Education India.

Dimitri Kartsaklis, Ian Fan, Richie Yeung, Anna Pearson, Robin Lorenz, Alexis Toumi, Giovanni de Felice, Konstantinos Meichanetzidis, Stephen Clark, and Bob Coecke. 2021. lambeq: An efficient high-level python library for quantum nlp. *arXiv preprint arXiv:2110.04236*, pages 1–19.

Vicente Coelho Lobo Neto, Leandro Aparecido Passos, and João Paulo Papa. 2020. Evolving long short-term memory networks. In *Computational Science – ICCS 2020*, pages 337–350, Cham. Springer International Publishing.

Aamir Mandviwalla, Keita Ohshiro, and Bo Ji. 2018. Implementing grover's algorithm on the ibm quantum computers. In *2018 IEEE international conference on big data (big data)*, pages 2531–2537. IEEE.

Francesco Petruccione Maria Schuld. 2021. *Machine learning with quantum computers*. Springer.

Kosuke Mitarai, Makoto Negoro, Masahiro Kitagawa, and Keisuke Fujii. 2018. Quantum circuit learning. *Physical Review A*, 98(3):032309.

Lee J O'Riordan, Myles Doyle, Fabio Baruffa, and Venkatesh Kannan. 2020. A hybrid classical-quantum workflow for natural language processing. *Machine Learning: Science and Technology*, 2(1):015011.

Shyambabu Pandey, Nihar Jyoti Basisth, Tushar Sachan, Neha Kumari, and Partha Pakray. 2023. Quantum machine learning for natural language processing application. *Physica A: Statistical Mechanics and its Applications*, page 129123.

Shyambabu Pandey, Pankaj Dadure, Morrel VL Nunsanga, and Partha Pakray. 2022. Parts of speech tagging towards classical to quantum computing. In *2022 IEEE Silchar Subsection Conference (SILCON)*, pages 1–6. IEEE.

Frank Phillipson. 2020. Quantum machine learning: Benefits and practical examples. In *QANSWER*, pages 51–56.

Russell and Stuart J. 2010. *Artificial intelligence a modern approach*. Pearson Education, Inc.

Abdel-Nasser Sharkawy. 2020. Principle of neural network and its main types. *Journal of Advances in Applied & Computational Mathematics*, 7:8–19.

Jinjing Shi, Zhenhuan Li, Wei Lai, Fangfang Li, Ronghua Shi, Yanyan Feng, and Shichao Zhang. 2023. Two end-to-end quantum-inspired deep neural networks for text classification. *IEEE Transactions on Knowledge and Data Engineering*, 35(4):4335–4345.

Ryan Sweke, Frederik Wilde, Johannes Meyer, Maria Schuld, Paul K. Faehrmann, Barthélémy Meynard-Piganeau, and Jens Eisert. 2020. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum*, 4:314.

Advaitha Vetagiri, Prottay Kumar Adhikary, Partha Pakray, and Amitava Das. 2023. Leveraging gpt-2 for automated classification of online sexist content. *Working Notes of CLEF*.