Text Classification   
*Muhammad Hammad   
21k-3181*  
*Abdul Raheem   
21k-3951*  
Quantum Computing

***Abstract*—**

**This project explores sentiment analysis using an innovative approach that integrates quantum computing and machine learning. A Quantum Convolutional Neural Network (QCNN) is employed to classify and predict the sentiment of textual inputs. The system is built on a hybrid Flask-Django framework, combining the robustness of Django with the lightweight and scalable nature of Flask. Text inputs are processed through the web interface, sent to the quantum neural network for classification, and the sentiment prediction is returned to the framework for display. This project highlights the potential of quantum computing in natural language processing, offering a novel methodology for text classification tasks with applications in opinion mining, customer feedback analysis, and more.**

* Introduction

Sentiment analysis is a critical field in natural language processing (NLP), enabling applications such as customer feedback analysis, opinion mining, and automated content moderation. Traditional machine learning models and deep learning techniques have shown significant success in tackling sentiment analysis tasks, yet they are often constrained by computational demands and limitations in capturing complex patterns within textual data. Quantum computing, with its potential to solve certain problems exponentially faster than classical computing, has emerged as a promising frontier for enhancing machine learning methodologies.

This project leverages the principles of quantum computing to design a Quantum Convolutional Neural Network (QCNN) for text classification and sentiment analysis. By integrating quantum mechanics into deep learning, the QCNN can process information in ways that classical models cannot, potentially leading to improved accuracy and efficiency in identifying sentiment.

The system is built on a hybrid Flask-Django framework, combining the simplicity and scalability of Flask with the robustness and structured architecture of Django. This combination facilitates a user-friendly interface for input collection and a backend capable of managing data processing, quantum computations, and result delivery. Users input text through the interface, which is then processed by the QCNN to classify the sentiment as positive, negative, or neutral. The predicted sentiment is returned and displayed to the user, demonstrating a practical implementation of quantum-enhanced NLP.

This project not only explores the practical applications of QCNNs in sentiment analysis but also highlights the synergy between classical frameworks and quantum computing. By bridging these technologies, the system serves as a foundation for further research and development in quantum machine learning and its applications in NLP.

* *Methodology and Literature*

*The method section outlines the architecture and workflow of the system, detailing how data is processed and predictions are made.*

*System Design*

*The project integrates Flask and Django frameworks to provide a seamless user interface and backend functionality. Text input is collected via the web interface, preprocessed for compatibility with the Quantum Convolutional Neural Network (QCNN), and sent to the quantum backend for analysis.*

*Quantum Convolutional Neural Network (QCNN)*

*The QCNN model is implemented using quantum computing principles. It applies quantum gates and convolutional layers to extract features and classify the input text into positive, negative, or neutral sentiments.*

*Integration Pipeline*

*The text input passes through the following stages:*

*Input Collection: Text is entered through a user-friendly web interface.*

*Preprocessing: Text normalization, tokenization, and feature extraction are performed.*

*Quantum Model Processing: Preprocessed text is fed into the QCNN for sentiment classification.*

*Output Display: The sentiment result is returned to the Flask-Django framework and displayed to the user.*

*Technology Stack*

*Frontend: Flask for lightweight API endpoints and UI interaction.*

*Backend: Django for database management and robust application logic.*

*Quantum Tools: Libraries such as Qiskit for quantum computing operations.*

* *Results*

The Quantum Convolutional Neural Network (QCNN) achieved an accuracy of 90% in classifying text inputs. The model successfully predicts whether the entered text pertains to a food item or an IT-related item. Testing was conducted on a balanced dataset, and the QCNN demonstrated consistent performance across both categories, accurately identifying contextual and semantic cues to assign the appropriate labels. These results highlight the model's capability to process and classify diverse textual data effectively, showcasing the potential of quantum-enhanced approaches in specialized text classification tasks.

* Conclusions

This project demonstrates the potential of integrating quantum computing with natural language processing through the development of a Quantum Convolutional Neural Network (QCNN) for sentiment analysis. By leveraging the unique computational capabilities of quantum mechanics, the QCNN provides a novel approach to text classification tasks, offering insights into the future of quantum-enhanced machine learning. The hybrid Flask-Django framework ensures a practical and user-friendly implementation, enabling seamless interaction and efficient processing. This work serves as a foundation for exploring more advanced quantum models in NLP, paving the way for innovative applications in sentiment analysis and beyond.

References

Yixiong Chen and Weichuan Fang, Multi-Scale Feature Fusion Quantum Depthwise Convolutional Neural Networks for Text Classification, 2024.