Quantum Computing For AI and NLP

Ronit Jha, Yanin Charoenpornsawat, Ayomide JeJe, Shane Sparks, Damir Cavar*
NLP-Lab – https://nlp-lab.org/

Introduction

- AIs use embedding models (vectorized language, image, audio data)
- Quantum AI relies heavily on classical computing:
  - **Variational quantum circuits**
  - N-shot measures
- Problems:
  - How can we compute **non-linear activations** in **quantum environments**?
  - Is there any **quantifiable loss** from classical to quantum?
  - Are neural models viable within a **purely quantum state**?

Objective

- Explore the limits for purely quantum circuits for neural networks
- Can we reuse classical embeddings in Quantum Computers?

Methods

Requirements for a Neural Model in Quantum:

- **Vector Embeddings**
  - Complex Numbers
    - **Imaginary and Real parts**
    - Maps naturally into complex-valued quantum state space
  - Normalized
    - Enforces constraint for valid quantum state loading
  - Power of 2 length
    - Corresponds to 2^n-dimensional Hilbert space of n qubits
- **Linear Transformation:**
  - Quantum circuits naturally implement linear, unitary transformations
  - Act like **neural network weight matrices** with orthogonality and unitarity constraints
- **Nonlinear Transformation:**
  - Decoherence Gate
    - Utilizing **noise** within the quantum environment
  - Measurement
    - Collapses qubits from superposition to 0 or 1.
  - Parametric Rotation Gate
    - (Rx, Ry, Rz) introduce nonlinearity through trigonometric transformations of quantum states.
  - Dimension Reduction
    - **Encode across N qubits**, trace out N-1 qubits for nonlinear compression

Results

- **Code-base**
  - Python Module **NLQK** (https://nlqk.ai/)
  - **Qiskit** and **Pennylane**
- **Lexical Similarity Scores Classical and Quantum**
  - Classical Embeddings to Quantum States (GPT (**OpenAI**) & Claude (**VoyageAI**) Vectors)
  - Similarity scores (Simulators & IBM Quantum)
    - VoyageAI embedding correlation coefficient:

	Classic	Q Real	Q Comp.int.	Q Comp.split
Classic		0.9756	0.9755	0.9731
Q Real			0.9535	0.9543
Q Comp.int.				0.9570

  - **Methods for Embeddings:** Classical vs Quantum Real vs Quantum Complex
  - **666 Word Pairs** from SimLex-999
- **Corpora**
  - Lexical elements for semantic similarity measures
  - Vectors for a variety of embedding models
  - Hamiltonians for the quantum states representing lexical embeddings
- **CBIRD**
  - Implementation of a **Complex PyTorch** implementation of Google AI's 2018 BERT
    - Using GitHub **PyTorch BERT** for real vector embeddings
    - Adding complex nonlinearity functions
      - **ModReLU**
      - **CReLU**

Discussion and Future Work

- Train embeddings with CBIRD and run experiments on classical and quantum
- Validate semantic preservation by benchmarking similarity measures across both simulators and accessible quantum hardware
- Experiment with Variational Circuits for AI (text similarity, classification) using the CBIRD embeddings

References

Devlin et al. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Association for Computational Linguistics.

Cavar et al. (2025) Old Wine in New Bottles: Using Classical Word Embeddings in Gate-Based Quantum NLP Systems. Springer Nature.

Damir Cavar, Koushik Reddy Parukola (2025) Word and Text Similarity Using Classical Word Embeddings in Quantum NLP Systems. Quantum Machine Learning in Signal Processing and Artificial Intelligence at the 2025 IEEE.

Acknowledgements

- NLP-Lab:
  - Arnav Jain
- Funded in 2024/2025 as a Center for Quantum Technologies (CQT) project