The American Journal of Applied Sciences
ISSN 2689-0992 Volume 08 - 2026

Quantum Natural Language Processing for Next-Generation Intent Recognition: Foundations, Techniques, And Future Directions

Rohan Mandar Salvi
Independent Researcher, USA

Received: 02 Jan 2026 | Received Revised Version: 18 Jan 2026 | Accepted: 22 Feb 2026 | Published: 28 Feb 2026

Volume 08 Issue 03 2026 | Crossref DOI: 10.37547/tajas/Volume08Issue02-08

ABSTRACT

AI-based systems in virtual assistants, conversation agents, and automated platforms rely heavily on intent recognition in Natural Language Processing (NLP). Using TF-IDF with classic classifiers results in reliable baselines, although using major transformer models, such as BERT, offers a deeper analysis and requires many resources. New developments in Quantum Machine Learning (QML) suggest new architectures that might help solve language problems. This study examined different types of intent classification models. A model can be set up using (i) classical TF-IDF + Logistic Regression, (ii) transformer-based BERT, and (iii) a quantum-classical variational quantum circuit (VQC) model from PennyLane. The models were evaluated using a custom JSON dataset with many intent examples. The results indicate that the classical model achieved the best performance among all, generating an accuracy of 68.97% and an F1-score of 70.69%, compared to the performance of BERT (accuracy=65.52% and F1-score = 63.91%) and the VQC model (accuracy=31.03% and F1-score = 25.65%). In addition to explaining the differences in the performance of the models, the visualizations pointed out their sensitive reactions to the structure and level of data in the categories. This paper discusses the advantages and disadvantages of different models, the challenges of using current quantum simulation tools, and the usefulness of QNLP in real-time and privacy-sensitive applications. At this current step in quantum model building, the findings here provide important guidance for future efforts on explainable, scalable, and hardware-accelerated QNLP.

Keywords: *Quantum NLP, Intent Recognition, Variational Quantum Circuit, TF-IDF, BERT, PennyLane, Transformer Models, Quantum Machine Learning, Natural Language Understanding, Hybrid Architectures*

Cite This Article: Salvi, R. (2026). Quantum Natural Language Processing for Next-Generation Intent Recognition: Foundations, Techniques, And Future Directions. The American Journal of Applied Sciences, 8(2), 69–80. https://doi.org/10.37547/tajas/Volume08Issue02-08

1. Introduction

NLP has significantly enhanced human communication with computers by enabling machines to comprehend, read, and respond to human language [1]. Identifying the intent expressed by users is a critical application of NLP that supports chatbots, voice assistants, smart customer service, and autonomous agents [20, 24]. Recent research on multi-agent conversational AI architectures and omnichannel search systems has further underscored the importance of maintaining context and consistency across diverse interaction channels. With proper intent recognition, machines behave according to the current situation, improving task automation and user experience. Recognizing intent can be achieved in various ways, ranging from basic systems involving rules to advanced statistical and machine learning algorithms, each bringing incremental improvements in pattern detection. The use of transformers, such as BERT, has helped increase the detection of intent across different domains.

Despite progress, there are still some obstacles in recognizing strong intent. The messages users write in

real life are often not specific, have similar meanings, and depend on the situation. Based on the context, a question like "Can I reset my password?" may require assistance, necessitate action, or relate to security. [2] It is often difficult for classical NLP to work on subtle details, especially in settings where data are either scarce or noisy. Using deep learning, overfitting datasets dominated by a few types of data and not easily interpretable prevent them from being deployed in sensitive systems. Additionally, advanced models are too complex to be deployed on devices that require lightweight computing or quick processing. Additional approaches are needed to explain linguistic ambiguity and effectively organize high-dimensional meanings.

Quantum computing employs a novel method that utilizes qubits, entanglement, and superposition, which could lead to new opportunities for machine learning [3]. QML algorithms, such as VQCs and QNNs, may identify and generalize patterns in data that surpass the capabilities of conventional computing models [4]. This approach is particularly relevant for NLP because tokens and their embeddings often reside in high-dimensional and sparse regions of the vector space. QNLP is intended to support language-based reasoning, classification, and sequence modelling using quantum models. Although hardware is still a cause of delay, using quantum layers in traditional artificial neural network simulators is now a practical approach to exploring the advantages of quantum computing. The main motivation for this research is the potential improvements QML could bring to accuracy, ability to handle diverse situations, and robustness of intent recognition.

This study explores and compares classical, deep learning, and quantum-enhanced models in their recognition of intent through experiments and architecture. We implemented three distinct approaches: (i) a TF-IDF + Logistic Regression model, (ii) a BERT-based classifier with transformer layers, and (iii) a PennyLane-assisted model that uses quantum circuits designed for classification. It relies on a JSON dataset with different intent labels and compares the models by examining their accuracy and F1-score. The aim was to test the performance of QML systems in restricted settings and determine whether they effectively handle ambiguous or competing intent cases.

2. Related Work and Background

2.1 Classical Approaches to Intent Detection

Initially, rule-based systems and keyword comparisons were the primary methods for intent recognition. They used handwritten rules to categorize user intents using pattern-matching templates or regular expressions. Although straightforward rule-based systems are accurate in narrow settings, they may fail to perform well when dealing with diverse language types or ambiguous expressions. To address these issues, statistical and machine learning models began to use vector space models of text. TF-IDF is one of the most popular methods for transforming textual information into numerical vectors that represent the weight of a word based on its frequency in a group of documents [5]. TF-IDF is easy to compute and provides strong features when working with classifiers such as LR, SVM, and Naïve Bayes [6].

These ML models utilize labelled data to discover decision boundaries and have proven to be comparable to other solutions in small-to medium-sized tasks involving intent recognition [7]. However, they are not strong enough to identify the subtleties in meaning and connections between different words, particularly with long or ambiguous phrases. Additionally, traditional machine learning classifiers generally struggle with class imbalance and sentences with different contexts because they view the input text as a collection of words without considering the order in which they appear. TF-IDF and lightweight classifiers work well as baselines, supporting the testing of new and complicated models in places with limited resources [8]. Notably, clustering-based approaches have also been explored for multivariate and spatio-temporal data analysis, with studies evaluating traditional methods such as DBSCAN and K-Medoids alongside deep learning-based clustering algorithms for high-dimensional datasets [19, 21, 22, 23].

2.2 Transformer-Based Models (BERT)

The release of transformer-based models, such as BERT, has significantly shifted the approach to NLP [9]. With the aid of a multi-layered bidirectional transformer encoder, BERT can learn contextual word embeddings that recognize connections between words and sentences in various ways [10]. BERT was trained using a large amount of data without needing labelled examples; therefore, it can be adjusted for things like intent classification using only a little more training information [11]. BERT has proven to be better for intent recognition because of its ability to handle long sequences, different meanings of the same word, and maintain the context intact [12]. The final output of the

BERT models is then fed to a classification head so that the model can process the raw input and predict the intent.

BERT and other transformer models have certain limitations. They require powerful computers to handle both the training and execution. Furthermore, these models are considered black boxes, making it difficult to understand their inner workings and interpret their results. When an application is sensitive, a lack of transparency can be highly dangerous. Additionally, because BERT requires a huge amount of data to learn and is then fine-tuned, it is not very useful for simple applications or those working in real time. This weakness has led scientists to explore quantum machine learning models and similar approaches, which may be more effective in these areas.

2.3 Quantum Computing for NLP

Quantum computing relies on concepts from quantum mechanics, including superposition, entanglement, and quantum interference, for information processing [13, 14]. Classical bits either have a 0 or a 1 value, but qubits can be in different states simultaneously, allowing for meaningful computations to be performed much faster and more efficiently. This shift provides more options for machine learning, particularly in domains where the data are very high-dimensional and structured in a specific way. Because VQCs are compatible with current quantum technology, they have become increasingly popular in the field of QML. The circuits are adjusted by tuning the gates and fine-tuning them using classical techniques that involve gradients. To perform NLP tasks on quantum computers, classical text is first encoded into quantum states using methods such as basis, amplitude, and angle encoding [15, 16]. Angle encoding is useful in low-dimensional quantum circuits because it enables the representation of continuous inputs through the rotation of qubit gates.

Several approaches can help create quantum-enhanced NLP models. PennyLane, a framework from Xanadu, facilitates the use of a combination of quantum and classical approaches by integrating it with PyTorch and TensorFlow [17]. It offers layers that enable the building of quantum circuits, partitioning them, and applying regular optimizers for training the models [18]. PennyLane is equipped with QNodes that establish a connection between quantum devices or simulators and classical systems, allowing quantum computing to be used for tasks such as classification, clustering, and word meaning comparison. Although Noisy Intermediate-Scale Quantum (NISQ) hardware has few and low-stability qubits, it can still be used to develop and test QML algorithms in simulated environments. Although QNLP is still in its early stages, initial studies have shown that integrating different technologies can equal or surpass the performance of more traditional baselines, particularly in cases of ambiguity and when dealing with small datasets.

2.4 Gap in Literature: Lack of Hybrid Comparative Studies

Only a few experimental studies have compared classical, deep learning, and quantum-based methods for intent recognition, despite the increased interest in Quantum Machine Learning (QML). Most prior efforts have been dedicated to theory or to testing quantum models in isolation, rather than comparing their results to better-known models such as TF-IDF and BERT. It has also been found that most data used for research in intent recognition consist of simple words and do not reflect real-life language diversity. Additionally, there is a missing piece in ready-to-use systems that demonstrate how to utilize quantum tools in Natural Language Processing (NLP), such as PennyLane. Few studies have examined the application of interpretable and scalable hybrid quantum-classical systems for NLP-related work. While evaluations of clustering algorithms for spatio-temporal data have demonstrated the value of systematic comparative analyses across different algorithmic paradigms [21, 23], similar rigorous comparisons remain scarce in the quantum NLP domain. This study addresses these issues by testing TF-IDF, BERT, and quantum models on the same intent classification task. The use of similar data and standard indicators in this study provides valuable insights into how systems for question answering in Natural Language Processing compare.

3. Methodology

The dataset, preprocessing, model architectures, and training settings are discussed in this study. The goal of this study was to review the differences between intent recognition based on a machine learning setup, a neural network, and a mixed quantum-classical model. The experiments used the same JSON-formatted dataset to enable a fair comparison.

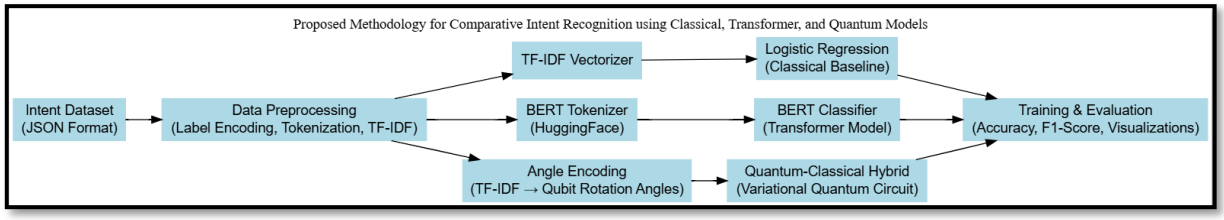

Figure 1: Proposed Methodology Framework

The methodology diagram outlines a simple method for intent recognition using classical, transformer, and hybrid quantum-classical approaches. An intent dataset written in JSON was used as input, which then underwent a unified step of encoding labels, making TF-IDF vectors, and tokenising according to BERT. Based on the shared model, the pipeline now branches into three separate modelling branches. First, TF-IDF is run to vectorize the terms, and then Logistic Regression is applied to serve as the reference. On the second path, the inputs are tokenized using BERT, and a transformer model is adjusted to perform contextual classification. In this method, TF-IDF features are transformed into quantum angles and passed on to a VQC using joint integration with PyTorch, as offered by PennyLane. They all share a similar phase, where the results are judged by accuracy, F1-score, and visual analysis. This architecture allows for the benchmarking of traditional, deep learning, and quantum computing systems using a similar experimental approach.

3.1 Dataset Description

The data analyzed for this study were sent from Chatbots: Intent Recognition Dataset contains a range of intent categories, and for each category, there are associated natural language utterances. Under the key "intents" was a list of objects, and each object included a "tag" for the intent class, examples of "patterns" given by users, and downstream "responses" for use after training, but not during learning. An example of this would be an object that represents "greeting" and uses "Hi," "Hello," and "Good morning" as patterns. Information for supervised learning was gathered by taking the "patterns" and "tag" fields only. Each pattern example was used as an input, and its tag was regarded as the real label. The data that were finally used included several hundred labelled sentences, all of which fit into approximately 10 to 15 different intent groups. Many of the intents could be used for general reasons ("goodbye," "greeting") or a particular task ("order_status," "cancel_item").

The labels were plotted on a graph to determine whether there were balanced classes. The analysis highlighted the typical unevenness present in actual datasets. Some intents, such as "greeting" and "thanks," were used significantly more often than others, including "technical_issue" and "return_policy." Equal class representation was achieved at a later step by following the process of stratified training-test splitting for evaluation.

3.2 Data Preprocessing

Preprocessing was uniformly performed across all three models to ensure consistency. However, the preprocessing method varied slightly depending on whether the model used a classical, transformer, or quantum-classical approach. For the TF-IDF and Logistic Regression models, the text data were vectorized using the TfidfVectorizer module from scikit-learn. Each sentence was converted into a simple numerical matrix by calculating the frequency of occurrence of a given token in the files and its commonality in the entire collection. The vectoriser was set up with max_features=1000 to prevent too many features from being included, and ngram_range= (1, 2) to note both single and double words. Noise-inducing words were deleted by default to enhance data clarity.

BERT's preprocessing was handled using HuggingFace's BertTokenizer. The encoded_plus method converts raw text into tokens, attention masks, and token type IDs. The sequences in batches were all shortened or lengthened to 32 tokens each. The tokenized outputs were used to set up DataLoaders in PyTorch, facilitating the efficient training of batched mini-batches during the fine-tuning. All intent labels (tags) were assigned an integer value using LabelEncoder in scikit-learn, so every unique tag had a unique number. Each model used the same numbers, allowing the target variables to be consistent in all training and testing. Finally, the dataset was divided into a training set, taking

80%, and a test set covering 20%. The categories were split using labels for intent to maintain their distribution in the two subsets. Therefore, all models were trained with data from all intent categories, and the test set provided a balanced way to test the models with minority classes.

3.3 Model Architectures

Three different model architectures were developed and applied for classic, deep learning, and quantum intent recognition. The classical pipeline involved converting documents into TF-IDF vectors and then using Logistic Regression. After converting the input texts into high-dimensional TF-IDF matrices, they were processed using the LogisticRegression model offered by scikit-learn. The function was trained with a regularization parameter C=1.0, using the lbfgs solver. Among the models, Logistic Regression distinguished itself owing to its ease of understanding and performance on small pieces of text. Using the frequency of tokens in texts, the model was able to classify categories based on their typical keywords.

For the second approach, a BERT classifier was created using the BertForSequenceClassification model from HuggingFace. A softmax activation was used after a fully connected dense layer on the pooled output of the [CLS] token to fine-tune the BERT-base-uncased base model. Fine-tuning was performed for 3 to 5 iterations, and overfitting was prevented by setting up early stopping. The model's success is due to the use of transformers to recognize both the style and meaning of sentences, despite the TF-IDF model not having similar abilities. The quantum-classical hybrid architecture was the third model used, and it was combined with PennyLane and PyTorch. Data from normalized TF-IDF vectors or reduced embeddings were first converted into quantum states by encoding the results in terms of angles. Here, the encoding strategy was based on choosing rotation angles on a single qubit for each feature value to obtain a quantum-ready input. Simulator restrictions required the feature vector to have between four and six dimensions, the same as the number of qubits.

PennyLane's default.qubit simulator was used to design the VQC. The circuit was built using three main components. The first layer works with the angle version of the data, the middle layer includes rotational and entanglement gates to represent parameters, and the last layer measures the expected values. The number of qubits is limited to four and is only allowed for two or three layers in the circuit to maintain both expressiveness and the ability to simulate the circuit. The quantum circuit was enclosed in a QNode and converted into a differentiable layer using the torchlayer package. This layer was added to an ordinary feedforward network constructed using PyTorch. The backpropagation algorithm was used to train the hybrid architecture, as gradients were moved backward along the quantum graph from the classical loss using PennyLane's automatic differentiation module.

3.4 Training Configuration

The code was run in Google Colab to ensure that all the models ran the same and our results were consistent. There were not enough resources for training the BERT model in batches larger than 16, but TF-IDF and quantum models could process batches of 32 samples. BERT and quantum models were optimized with the Adam optimizer, and they had learning rates of 2e-5 and 0.01, respectively. When wrapped into a PyTorch pipeline, Logistic Regression used the lbfgs solver or an Adam-based approach. It uses the categorical cross-entropy loss function for training, which was developed for multi-class classification tasks. The quantum model was wrapped in a custom manner to change the expectation values into logits that can be handled by PyTorch's loss functions. The quantum simulations used the default settings of PennyLane. Qubit backend, which has four qubits. Because quantum computers were not accessible, the computations were performed on a computer using a virtual simulator. To obtain the same results each time, the random seeds were fixed for NumPy, PyTorch, and PennyLane. The design chosen for this method makes it easier to examine all the different approaches to intent recognition.

4. Experimental Evaluation and Results

4.1 Evaluation Metrics

To evaluate and compare the models under different architectures, the accuracy and F1-scores were calculated. Accuracy helped measure the overall successful predictions, and the F1-score, an average of precision and recall, focused on how well the model performed with minority class data. Given that one class was much rarer than the others in the dataset, it was necessary to determine the accuracy of each model on rare intent classes by measuring the F1-score. The metrics were calculated on the test data, which comprised 20% of the dataset, to ensure that the evaluation was fair

and the same for every model. Resource implications were monitored by checking the time required to train and use each model, paying special attention to the BERT model and the hybrid architecture. The model using TF-IDF and Logistic Regression trains very quickly, but BERT and the quantum method are much slower and require more time to be trained, regardless of the size of the data.

4.2 Exploratory Data Analysis

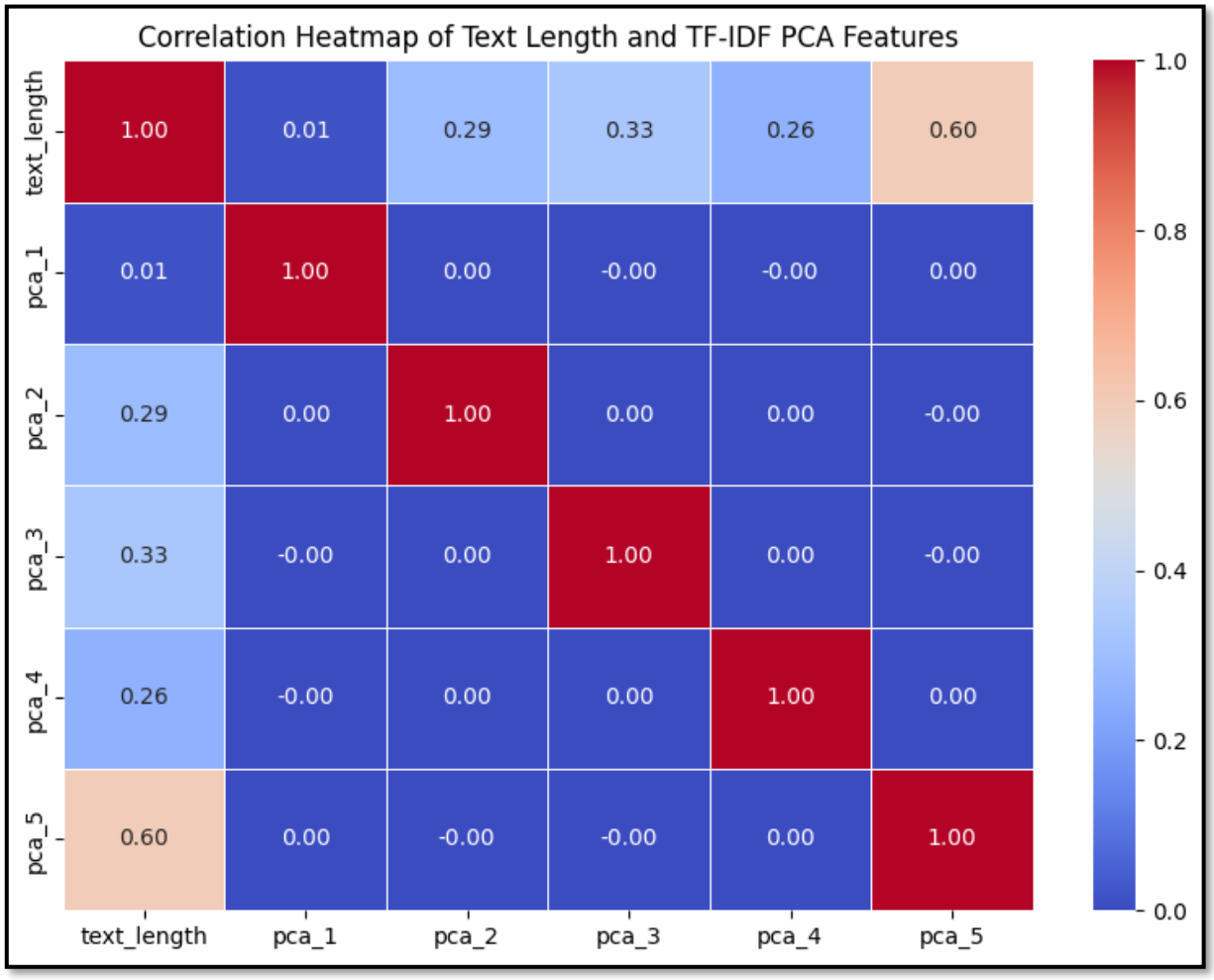

Figure 2: Correlation Heatmap of Text Length and TF-IDF PCA Features

Figure 2 demonstrates the Pearson correlation between the number of words in the text and five main TF-IDF vectors. A moderate correlation (r = 0.60) existed between sentence length and the fifth principal component (pca_5), suggesting that longer sentences may contain richer terms that influence this component of information. Because the correlations with other elements were low, the PCA dimensions reflected unique, distinct semantic concepts, primarily independent of sentence length. This confirms that the process did not affect the range of lexical terms or lead to issues with multicollinearity owing to the input length.

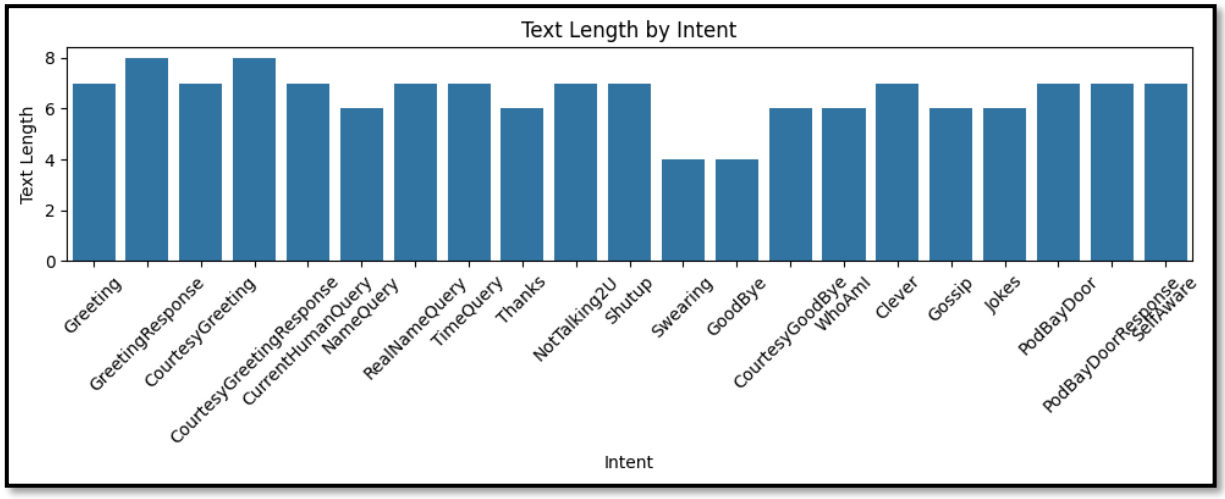

Figure 3: Text Length by Intent

Figure 3 shows the average sentence length per intent category. GreetingResponse and CourtesyGreetingResponse intents often have longer sentences because people are polite. Otherwise, quick and blunt comments are the main characteristics of swearing and goodbye intents. The analysis highlights differences in language complexity within social classes, which is crucial for obtaining accurate model results. Models that do not consider sequence, such as TF-IDF and Logistic Regression, may struggle to handle intents where the meaning is formed through the interaction of several words.

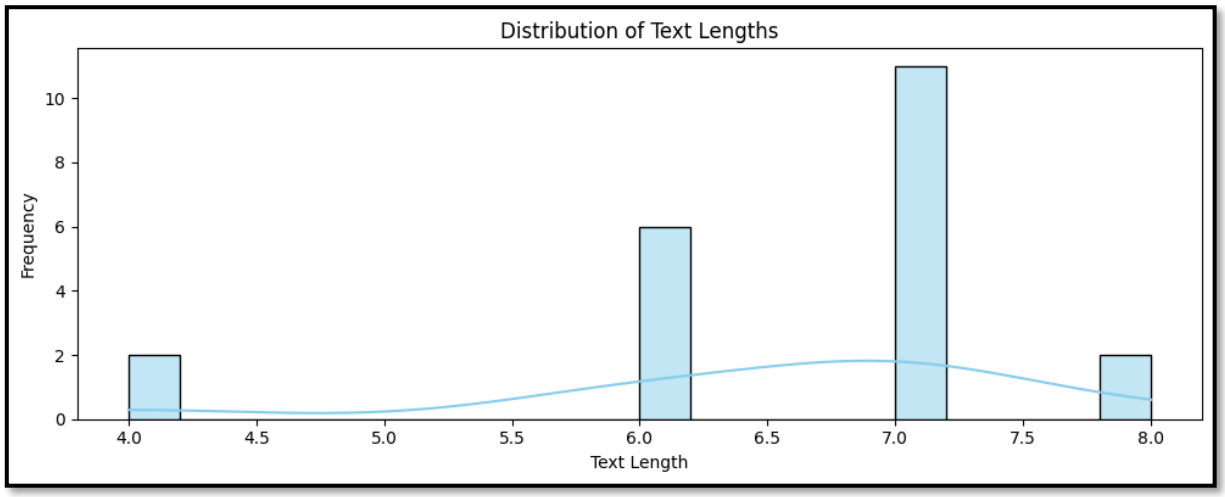

Figure 4: Distribution of Text Lengths

Figure 4 shows the distribution of the text lengths in the dataset. Most of the time, people's utterances range from six to seven tokens, while there are fewer at the beginning or end of the scale. Because the data show a consistent input size, it is fair to use a fixed padding length (32 tokens) for the BERT model. This further indicates that, despite different intent lengths, there is a main style that allows for easier group optimization and prevents the software from running out of memory during training.

4.3 Model Evaluation

```
=== Classical Model: TF-IDF + Logistic Regression ===
Accuracy: 68.97%
F1 Score: 70.69%
```

Figure 5: Logistic Regression Performance

Figure 5 illustrates that the TF-IDF + Logistic Regression model achieved an accuracy of 68.97% and an F1-score of 70.69%. These results emphasize that the model is powerful despite its simplicity and lack of contextual information. As the F1-score indicates, the model effectively handled imbalanced classes and identified common keyword patterns, albeit with potential difficulties in interpreting terms with multiple meanings.

```
Epoch 1 Loss: 3.1234
Epoch 2 Loss: 3.0647
Epoch 3 Loss: 2.9178
Epoch 4 Loss: 2.7885
Epoch 5 Loss: 2.7480
Epoch 6 Loss: 2.7300
Epoch 7 Loss: 2.6317
Epoch 8 Loss: 2.5329
Epoch 9 Loss: 2.5051
Epoch 10 Loss: 2.4842
=== Enhanced BERT-Based Classifier ===
Accuracy: 65.52%
F1 Score: 63.91%
```

Figure 6: BERT Classifier Loss and Scores

Figure 6 shows a plot of the epoch-wise loss, which demonstrates that the BERT model gradually improved as it was fine-tuned. Nevertheless, the performance of the top model at the end did not meet our expectations, with an accuracy of 65.52% accuracy and 63.91% F1-score. The reason for the underperformance could be overfitting, the small dataset size, or an inability to separate classes. Even with advanced models like BERT, it appears that they require a significant amount of diverse data or fine-tuning to outperform traditional baselines on small tasks.

```
=== Quantum Hybrid VQC Model ===
Accuracy: 31.03%
F1 Score: 25.65%
```

Figure 7: Quantum Hybrid VQC Model Performance

Figure 7 displays the Hybrid Quantum-Classical model's performance accuracy of 31.03% and F1 score of 25.65%. This indicates the difficulties encountered when attempting to transform large language data into simple

quantum circuits (with four to six qubits). Although this approach is promising in theory, it remains underdeveloped and cannot compete with existing classical or deep learning methods used for multi-intent classification in simulation scenarios. Nevertheless, this demonstrates a potential opportunity for further studies on the large-scale implementation of QNLP.

4.4 Model Comparison

```
=== Comparative Performance Summary ===
                 Model  Accuracy  F1 Score
0  Logistic Regression  0.689655  0.706897
1                 BERT  0.655172  0.639080
2          Quantum VQC  0.310345  0.256486
```

Figure 8: Comparative Performance

Figure 8 presents a table that facilitates a comparison of the results of the three models. Logistic Regression yielded the best accuracy and F1-score, followed by BERT, and then the Quantum VQC model (Figure 8). The table indicates that when there are few samples, tuning a simple model outperforms a more advanced model. It also highlights that, at this point, the quantum model is limited, stressing the significance of ongoing changes in the design and access to more powerful quantum systems.

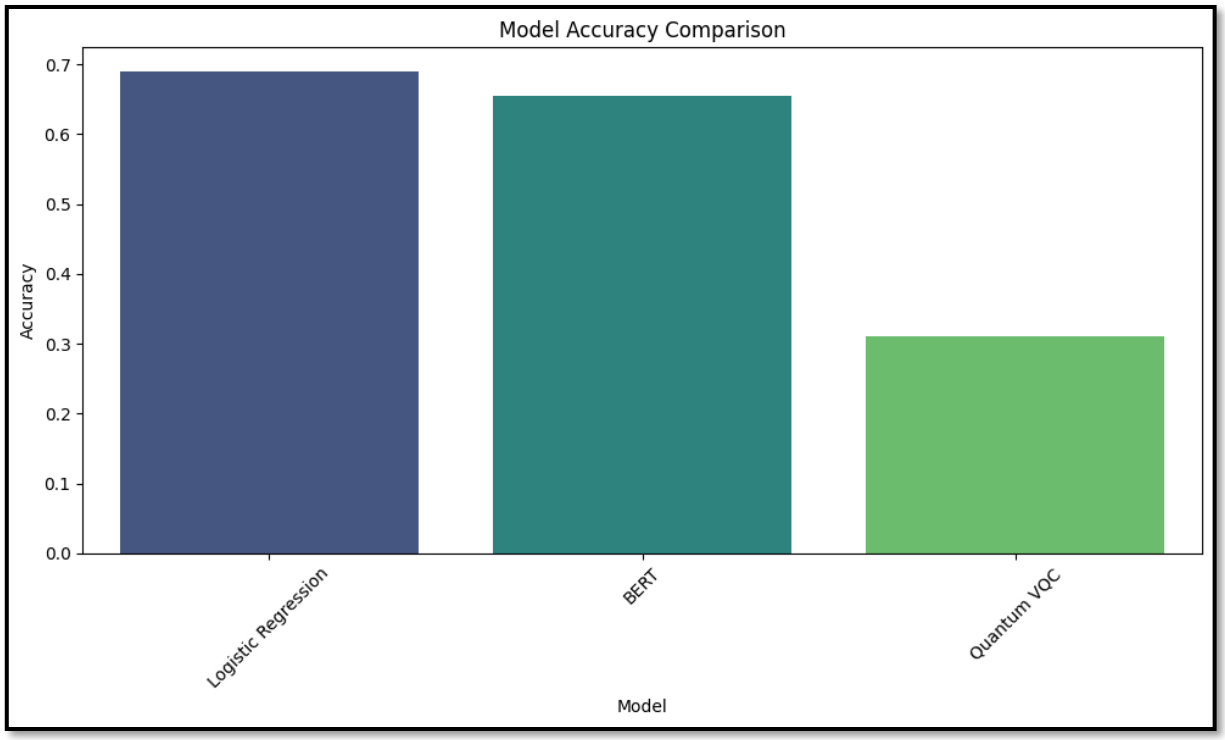

Figure 9: Model Accuracy Comparison

Figure 9 confirms the ranking of the three models based on performance. Logistic Regression performed the best, followed by BERT, and Quantum VQC ranked last. The figure reveals the experimental findings and is a useful tool for comparing the main approaches. This difference in results highlights that there is still a long way to go before quantum simulation can be applied to NLP tasks.

5. Discussion

5.1 Comparative Insights

Testing revealed clear downsides to efficiency when a model achieved a higher performance. The TF-IDF + Logistic Regression model demonstrated the best accuracy and F1-score compared to the other two models. This is noteworthy because it shows that classical NLP methods can still achieve strong results on structured and medium-sized tasks. In addition to being easy to understand and quick to calculate, Logistic Regression can handle resource limitations better than other models. However, the performance of the BERT model was lowered because it faced issues of overfitting and used the same data repeatedly. Although the quantum-classical model is innovative, it still scored the lowest because of the architectural and simulation difficulties. Basic models, such as RNNs, are better at detecting intent for smaller projects, whereas transformers require additional fine-tuning and better data to work effectively. Although quantum-enhanced models are currently showing limited results, they could enable more advanced innovation if qubits improve and encoding becomes smarter.

5.2 Practical Implications

For practical applications, logistic regression can be applied immediately to edge devices or mobile platforms, where speed and simplicity are crucial. BERT provides excellent results but works best on systems equipped with GPUs and is intended for large server applications. Owing to its reliance on simulation software, the quantum model cannot be used in real-world settings. Computing the results of these models is time-consuming, and the cost of training increases with the number of qubits. Such platforms are suitable for performing inference on advanced NLP tasks, especially when there are insufficient resources in traditional hardware.

5.3 Quantum Model Challenges

Implementing the Quantum VQC model is not easy because of various technical difficulties. Transforming classical data into quantum text embeddings requires many additional resources. Although angle encoding simplifies and differentiates the problem, it limits the number of qubits for input features, which can result in a loss of data or simple representations. In addition, present-day simulations do not use quantum speedup and depend on classical technology. The main challenge is related to scaling: the ability of a computer model increases exponentially as its complexity increases, even if the simulation is virtual. They underline that the use of QNLP models in practice is not yet as powerful as it could be, calling for updates and mixed methods, and requiring systems with actual quantum processors.

6. Limitations and Future Directions

6.1 Current Constraints

This study has limitations that make it difficult to generalize and understand the results. The dataset used for training had few samples and was limited in scope, including approximately 15 intent classes. At the prototyping stage, it was fine, but the lack of diversity and data might have slowed the learning of deep and quantum models. Further investigation of a larger dataset could highlight new or different patterns compared to those found in this study. All the tested quantum computations were run on simulators using PennyLane's default qubit. Although testing and experimentation are possible with simulated quantum circuits, they are much slower than real quantum computers. Moreover, they do not accurately represent the effects of decoherence, entanglement, and calibration of real-world quantum devices. Consequently, the time required for training, the reliability of the training, and the shape of the gradients can be quite different on real hardware. In addition, the study considered only a few intent classes, which makes it difficult to extend the findings to wider NLP issues.

6.2 Future Work

To address these issues, future studies should experiment with models on actual quantum devices, such as IBMQ, IonQ, or Rigetti. To understand the speed, capacity, and limits of present-day QPUs, one may use the AWS Braket or Azure Quantum platforms to access quantum computers. Furthermore, to test how well these systems function, experiments can be conducted on datasets that include data in many different languages. QFL for intent recognition has shown positive results, as it allows local quantum nodes to join forces in training a global model while keeping data private. Recent work on federated query rewriting for conversational AI has demonstrated the viability of privacy-preserving, cross-channel retrieval approaches that could complement quantum federated architectures [25]. The paradigm could be very helpful for companies involved in healthcare or finance, as it helps sort conscious and non-conscious information. Such transfer learning, as seen in BERT, may enable the collection of data that could improve the general

performance of QNNs. Finally, quantum models are still not well understood in most applications. More efforts are needed to create methods for understanding QNLP, such as visualizing the progress of quantum states and exploring the influence of entanglement on their features. It is vital to ensure that quantum-enhanced systems are explainable, given that creating trust in them is important wherever safety is crucial.

7. Conclusion

The main aim of this study was to examine the performance of classical, transformer-based, and quantum-classical hybrid systems for intent recognition under a genuine NLP setting. To test the algorithms, three specialized JSON pipelines for custom data were developed and used. The first is TF-IDF + Logistic Regression, followed by BERT and a VQC model created using PennyLane. The assessment of each model included common metrics—accuracy, F1-score, and training loss—besides assessments on how easy it would be for users to interpret and apply the model. It was found that when the amount of data is scarce, classical machine learning tends to yield better results than deep learning models in NLP tasks. Although BERT can process many contexts, it only demonstrated moderate gains and had somewhat high resource needs and a tendency to overfit the data. Owing to technical issues and encoding rules, the quantum model cannot compete with existing machine learning algorithms. The findings led to the identification of several major concepts. First, a model must be practical for a given task, not only based on its theory but also on factors such as scale and computing resources. Nevertheless, quantum NLP remains in its emerging phase. However, it may greatly benefit future research when combined with hardware support and innovative changes in architectures. Overall, comparing the designs highlights how simple models and those aware of data play a crucial role in the selection of intent-aware systems. Quantum Natural Language Processing (QNLP) is a recent development in natural language processing. Although these models are still being tested, they could play a major role in changing the usual NLP workflow, mainly in the areas of parallelism, security, and generalization. When quantum hardware and hybrid algorithms advance, QNLP is expected to move from being an interesting concept to a useful practice, impacting the field of human-computer language use. This study serves as a foundation for progress and highlights important areas that require future attention and development.

Referenecs

1. Harvey W. How Artificial Intelligence is Improving Human Communication with the Processing of Natural Language. EPH-International Journal of Science and Engineering. 2024 Dec 2;10(3):56-76.
2. Al Sharou K, Li Z, Specia L. Towards a better understanding of noise in natural language processing. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021), 2021 Sep (pp. 53-62). Available: https://aclanthology.org/2021.ranlp-1.7/
3. Taghandiki K. Quantum Machine Learning Unveiled: A Comprehensive Review. Journal of Engineering and Applied Research. 2024 Oct 1;1(2):29-48. https://doi.org/10.48301/jear.2024.446673.1021
4. Tomar S, Tripathi R, Kumar S. Comprehensive Survey of QML: From Data Analysis to Algorithmic Advancements. arXiv preprint arXiv:2501.09528. 2025 Jan 16. https://doi.org/10.48550/arXiv.2501.09528
5. Xiang L. Application of an Improved TF-IDF Method in Literary Text Classification. Advances in Multimedia. 2022;2022(1):9285324. https://doi.org/10.1155/2022/9285324
6. Nafis NS, Awang S. An enhanced hybrid feature selection technique using term frequency-inverse document frequency and support vector machine-recursive feature elimination for sentiment classification. IEEE Access. 2021 Mar 26;9:52177-92. https://doi.org/10.1109/ACCESS.2021.3069001
7. Mehdiyev N, Mayer L, Lahann J, Fettke P. Deep learning-based clustering of processes and their visual exploration: An Industry 4.0 use case for small and medium-sized enterprises. Expert Systems. 2024 Feb;41(2):e13139. https://doi.org/10.1111/exsy.13139
8. Saleem S, Hasan N, Khattar A, Jain PR, Gupta TK, Mehrotra M. DeLTran15: A deep lightweight transformer-based framework for multiclass classification of disaster posts on X. IEEE Access. 2024 Oct 11.
9. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I. Attention is all you need. Advances in Neural Information Processing Systems. 2017;30. https://doi.org/10.48550/arXiv.1706.03762

10. Warikoo N, Chang YC, Hsu WL. LBERT: Lexically aware Transformer-based Bidirectional Encoder Representation model for learning universal bio-entity relations. Bioinformatics. 2021 Feb 1;37(3):404-12. https://doi.org/10.1093/bioinformatics/btaa721
11. Prabhu S, Mohamed M, Misra H. Multiclass text classification using BERT-based active learning. arXiv preprint arXiv:2104.14289. 2021 Apr 27. https://doi.org/10.48550/arXiv.2104.14289
12. Raju A, Raju C. Advancing AI-driven customer service with NLP: A novel BERT-based model for automated responses. International Journal of Scientific Research in Computer Science, Engineering and Information Technology. 2024.
13. Khrennikov A. Roots of quantum computing supremacy: superposition, entanglement, or complementarity?. The European Physical Journal: Special Topics. 2021 Jun;230(4):1053-7. https://doi.org/10.1140/epjs/s11734-021-00061-9
14. Hughes C, Isaacson J, Perry A, Sun RF, Turner J. What is a qubit?. Quantum computing for the quantum curious. 2021:7-16. https://doi.org/10.1007/978-3-030-61601-4_2
15. Nausheen F, Ahmed K, Khan MI. Quantum Natural Language Processing: A Comprehensive Review of Models, Methods and Applications. arXiv preprint arXiv:2504.09909. 2025 Apr 14. https://doi.org/10.48550/arXiv.2504.09909
16. Zollner JM, Walther P, Werner M. Satellite image representations for quantum classifiers. Datenbank-Spektrum. 2024 Mar;24(1):33-41. https://doi.org/10.1007/s13222-024-00472-7
17. Ganguly S. Deep Quantum Learning. In: Quantum Machine Learning: An Applied Approach: The Theory and Application of Quantum Machine Learning in Science and Industry. 2021 Jul 28 (pp. 403-459). Berkeley, CA: Apress. https://doi.org/10.1007/978-1-4842-7098-1_11
18. Chen KC, Li X, Xu X, Wang YY, Liu CY. Multi-GPU-Enabled Hybrid Quantum-Classical Workflow in Quantum-HPC Middleware: Applications in Quantum Simulations. arXiv preprint arXiv:2403.05828. 2024 Mar 9. https://doi.org/10.48550/arXiv.2403.05828
19. Faruque O, Nji FN, Cham M, Salvi RM, Zheng X, Wang J. Deep spatiotemporal clustering: A temporal clustering approach for multi-dimensional climate data. In: ECML PKDD 2023. Lecture Notes in Computer Science, vol 14175. Springer, Cham. 2023. https://doi.org/10.1007/978-3-031-43430-3_6
20. Salvi RM, Barman PK. Evolving Architectures and Long-Horizon Planning in Multi-Agent Conversational AI: A Decade in Review. The American Journal of Interdisciplinary Innovations and Research. 2025;7.
21. Nji FN, Salvi RM, Tirumala S, Wang J, Zheng X. Evaluation of Traditional and Deep Clustering Algorithms for Multivariate Spatio-Temporal Data. Lawrence Livermore National Laboratory (LLNL), Livermore, CA (United States). 2024.
22. Salvi RM. Spatio-Temporal Multivariate Weather Data Clustering Using DBSCAN and K-Medoids Methods. University of Maryland, Baltimore County. 2023.
23. Nji FN, Salvi RM, Tirumala S, Wang J, Zheng X. Evaluation of Clustering Algorithms for Spatio-Temporal Multivariate Weather Data. Lawrence Livermore National Laboratory (LLNL), Livermore, CA (United States). 2022.
24. Salvi RM. Omnichannel Conversational Search: Maintaining Context and Consistency Across Voice and Web Interfaces. International Journal of Applied Mathematics. 2025;38(8s):1100-1114. https://doi.org/10.12732/ijam.v38i8s.630
25. Salvi RM. Federated Query Rewriting for Conversational AI: Privacy-Preserving, Cross-Channel Retrieval on Voice and Web. International Journal of Computational and Experimental Science and Engineering. 2025. https://doi.org/10.22399/ijcesen.4297