Enhancing Speech Extraction And Speaker Recognition Through Machine Learning And Deep Learning Integration

Cao Hoai Sang[1], Thi Thanh Cong[1], Nguyen Minh Nhut[1], and Nguyen Dinh Thuan[1]

University of Information Technology, Ho Chi Minh City, Viet Nam
21522541@gm.uit.edu.vn, 21521897@gm.uit.edu.vn,
nhutnm.17@grad.uit.edu.vn, thuannd@uit.edu.vn

Abstract. In the age of digital communication and telework, the need for advanced conversation processing becomes increasingly important. This study proposes a system capable of speech separation and speaker recognition. The approach combines acoustic features such as MFCC, XVector, DVector and Wavelet with machine learning including HMM or deep learning models like RNN and QCNN. The system is trained and tested on a labeled conversation dataset. Experimental results demonstrate that the system achieves high accuracy in speaker discrimination and speech extraction.

Keywords: Speaker Recognition · MFCC · QCNN · HMM · XVectors · DVectors · Wavelet

1 Introduction

With the increasing prevalence of online meetings, virtual collaboration, and remote communication, the demand for accurate and real-time speech processing systems has become more urgent than ever. Traditional automatic speech recognition (ASR) systems primarily focus on transcribing spoken content into text. However, these systems often fall short in distinguishing between different speakers and in isolating clean speech signals from overlapping conversations or background noise.

This research addresses these limitations by proposing an integrated framework that enhances both speech extraction and speaker recognition. By leveraging a combination of acoustic feature representations—such as MFCC, Wavelet, X-Vector, and D-Vector—and advanced modeling techniques including Hidden Markov Models (HMM), Recurrent Neural Networks (RNN), and Quantum Convolutional Neural Networks (QCNN), our approach aims to significantly improve the performance and robustness of ASR systems in real-time multi-speaker environments.

The proposed method is especially relevant for applications requiring speaker-aware transcription, speaker diarization, and secure voice-based authentication, where both accuracy and adaptability are critical.

2 Related Work

Automatic speech recognition (ASR) has undergone significant development through the integration of classical signal processing techniques and modern machine learning models. In this section, we review prior work grouped by feature extraction methods and model architectures that are foundational or relevant to our proposed QCNN-HMM approach.

2.1 Feature Extraction Techniques

Mel-Frequency Cepstral Coefficients (MFCC) remain one of the most widely adopted features in ASR due to their ability to represent perceptually relevant characteristics of speech. Rabiner [8] demonstrated the effectiveness of combining MFCC with hidden Markov models (HMM) for robust speech modeling, establishing a foundational pipeline for traditional ASR systems.

Wavelet Transform-based features have also been explored for speech representation, especially under noisy or non-stationary conditions. Unlike MFCCs, wavelets provide multi-resolution analysis in both time and frequency domains. Studies such as those by Gupta et al. [4] and Wang et al. [15] reported improved noise robustness when integrating wavelet features into ASR systems.

X-vector and D-vector embeddings represent deep learning-based approaches to extract speaker-specific characteristics from audio. X-vectors, introduced by Snyder et al. [9], leverage DNN/TDNN structures to generate discriminative embeddings for speaker recognition tasks. Similarly, Wan et al. [14] proposed D-vectors trained with GE2E loss, enabling effective few-shot speaker verification. These embeddings are now core components in modern toolkits like Kaldi and SpeechBrain.

2.2 Model Architectures in ASR

Hidden Markov Models (HMM) have long served as the backbone of conventional ASR systems. The work by Rabiner [8] laid the groundwork for HMM-based modeling using algorithms such as Viterbi decoding. Subsequent research has shown that hybrid architectures combining HMM with neural networks (e.g., MLPs or RNNs) can further improve performance [7, 12].

Recurrent Neural Networks (RNN) and their variants, such as LSTM and GRU, address temporal dynamics in speech data by maintaining memory over time. RNN-based architectures have achieved substantial success in end-to-end ASR. Notably, Deep Speech [5] and the work of Graves et al. [3] demonstrated the viability of RNNs in replacing traditional HMM pipelines for sequence modeling.

Quantum Convolutional Neural Networks (QCNN) represent a novel approach that integrates quantum computing principles with deep learning. Introduced by Cong et al. [1], QCNNs aim to reduce parameter redundancy while enhancing representational capacity through quantum entanglement and measurement. While still in early stages of ASR application, work by Yang et al. [16]

illustrates the potential of QCNNs when integrated with classical models, particularly for structured signal classification.

3 Background

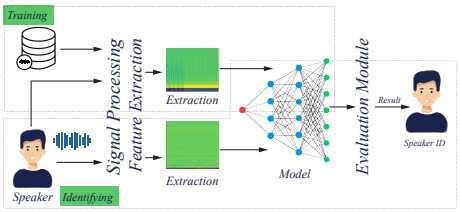

Fig. 1. Speaker Identification Flow

An Automatic Speech Recognition (ASR) system designed for speaker identification analyzes an input audio signal, denoted as $audio_x$. The ASR model, represented as $f(\cdot)$, aims to match the input with the most appropriate speaker profile based on the extracted voice features. In figure 1, when training or identifying the **feature extraction**, captures the acoustic characteristics of the input speech signal. Traditional systems often rely on **MFCC** to extract time-frequency representations of speech, while modern systems have adopted more advanced techniques, such as **XVector**, **DVector**, and **Wavelet**, which offer enhanced representations of speaker traits in different domains. In the next phase, **model processing**, the extracted features or raw spectrogram are fed into models such as **Hidden Markov Models (HMM)**, **Recurrent Neural Networks (RNN)**. These models map the features to an intermediate representation, enabling accurate speaker identification. [2, 3, 9, 13].

3.1 Hidden Markov Model (HMM)

HMM is a probabilistic model used to describe a sequence of observations $O = \{o_1, o_2, ..., o_T\}$ through a set of hidden states $Q = \{q_1, q_2, ..., q_N\}$. In ASR

An HMM is defined by three main parameters:

- $A = [a_{ij}]$: The state transition probability matrix, where $a_{ij} = P(q_{t+1} = j \mid q_t = i)$.
- $B = [b_j(o_t)]$: The observation (output) probability, where $b_j(o_t) = P(o_t \mid q_t = j)$.
- $\pi = [\pi_i]$: The initial state distribution, $\pi_i = P(q_1 = i)$.

In ASR, when a speech signal sequence is input, the goal is to find the optimal state sequence Q^* that maximizes the observation probability. When N models

for each speaker is created, the Viterbi algorithm is essential for finding the most likely state sequence based on the HMM [6]. To train the HMM, the Baum-Welch algorithm (a variant of Expectation-Maximization) is used to estimate the parameters.

3.2 Recurrent Neural Networks (RNNs)

Recurrent Neural Networks (RNNs) are widely used in ASR systems due to their ability to model temporal dependencies in sequential data. Unlike feedforward networks, RNNs maintain a hidden state that captures context from previous time steps, making them suitable for processing variable-length speech signals. Variants such as Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) have been shown to effectively capture long-range dependencies and improve recognition performance [3].

3.3 Mel Frequency Cepstral Coefficients (MFCCs)

Mel-Frequency Cepstral Coefficients (MFCC) are widely used in ASR systems to extract speech features by mimicking human auditory perception. The analog speech signal is first digitized, then passed through a high-pass filter and windowed to minimize noise and spectral leakage. The Discrete Fourier Transform (DFT) converts the signal to the frequency domain, which is then mapped to the Mel scale. A logarithmic function is applied to compress the dynamic range, and the Discrete Cosine Transform (DCT) converts this to the cepstral domain. Finally, first- and second-order derivatives (Δ, Δ^2) are added to capture temporal dynamics. [2]

3.4 Wavelet

In this work, we use the Discrete Wavelet Transform (DWT) to extract features from audio signals for speaker identification. Unlike the Fourier Transform, which only captures frequency content, the DWT provides both time and frequency information, making it suitable for analyzing non-stationary signals like speech.

Given an input signal, the DWT decomposes it into approximation and detail coefficients at multiple levels using a pair of low-pass and high-pass filters followed by downsampling. This hierarchical decomposition allows for capturing transient and localized features in speech.

We use the Daubechies-4 (db4) wavelet and apply the transform up to level 1. From each set of wavelet coefficients, we extract statistical features such as mean, standard deviation, max, min, median, and energy. These features serve as inputs for a Gaussian Hidden Markov Model (HMM) classifier or Recurrent Neural Network (RNN). [10]

3.5 X-Vector

The X-Vector model uses a deep neural network to generate speaker embeddings. It consists of:

- **Preprocessing**: Filters input features for localization.
- **Frame-level layers**: Convolutional or TDNN layers extract frame-level features.
- **Statistics pooling**: Aggregates features across the utterance using statistics (e.g., mean, standard deviation).
- **Speaker embedding**: Produces a fixed-length X-Vector representing speaker-specific information.

In this study, we use the pre-trained X-Vector model from the SpeechBrain library to extract speech features from audio segments. The X-Vector model is based on a deep neural network architecture, trained on large datasets: VoxCeleb to learn feature vector representations for individual speakers.

3.6 D-Vector

The **D-Vector** is a speaker embedding method that captures speaker-specific characteristics from short speech segments. It is typically extracted from the bottleneck or penultimate layer of a deep neural network trained for speaker classification. These embeddings are widely used in speaker verification, diarization, and voice cloning tasks due to their ability to represent speaker identity compactly and effectively.

As introduced by Variani et al. [11], the D-Vector system employs a deep neural network with a softmax output layer to classify speakers, and the output of an intermediate layer is used as the speaker embedding.

In this paper, we use the SpeechBrain's SpeakerRecognition class to load the pre-trained ECAPA-TDNN model. This model is designed to extract high-quality speaker embeddings. Then we used the Hidden Markov Models, Recurrent Neural Network, Quantum Convolutional Neural Network to perform training.

4 Combination Enhancement

We present our proposed models for speaker identification, which incorporate a novel Quantum Convolutional Neural Network (QCNN) architecture, along with a hybrid model that combines traditional Hidden Markov Models (HMM) and QCNN. Unlike conventional methods like RNNs or HMMs that rely on standard neural network architectures, our approach leverages quantum-enhanced processing to better capture both temporal and spectral features in speech signals.

4.1 Overview

The overall pipeline consists of the following stages: audio preprocessing, feature extraction (MFCC, Wavelet, X-vector or D-Vector), quantum encoding, QCNN modeling, and speaker classification. For the hybrid approach, we introduce a post processing stage using Hidden Markov Models (HMMs) to capture temporal consistency in speaker transitions. Figure **??** illustrates the overall system architecture.

4.2 Feature Extraction

We use some feature extractors such as Mel Frequency Cepstral Coefficients (MFCCs), Wavelet transform, X-Vectors, D-Vectors to extract features from raw audio segments corresponding to each speaker. In experiments with X-Vectors and D-Vectors, we use a pre-trained speaker model from SpeechBrain to capture high-level speaker characteristics.

4.3 Quantum Encoding and Circuit Design

This section represent the quantum feature mapping process. Our quantum circuit approach combines principles of quantum encoding with randomized parameterization. The circuit operates in two distinct phases:

Initial State Preparation We use Hadamard gate to init each qubit in a superposition state, then apply randomly parameterized rotation:

$$|\psi_{\text{init}}^{(i)}\rangle = RY(\theta_i) \cdot H|0\rangle$$

where $\theta_i \sim \mathcal{U}(-\pi, \pi)$ is sampled from a uniform distribution. This applies to all qubits $i \in \{0, 1, \ldots, n-1\}$ where $n = 7$ is the total number of qubits.

Entanglement Layer Adjacent qubits are entangled using CNOT gates, and also some additional random rotations are applied to target qubits:

$$|\psi_{\text{ent}}^{(i,i+1)}\rangle = RY(\phi_{i+1}) \cdot \text{CNOT}_{i,i+1}|\psi_{\text{init}}\rangle$$

where $\phi_{i+1} \sim \mathcal{U}(-\pi, \pi)$ is also sampled from a uniform distribution. This entanglement structure is applied sequentially for $i \in \{0, 1, \ldots, n-2\}$.

The complete quantum state preparation can be expressed as:

$$|\psi_{\text{out}}\rangle = \prod_{i=0}^{n-2} U_{\text{ent}}^{(i,i+1)} \prod_{i=0}^{n-1} U_{\text{init}}^{(i)} |0\rangle^{\otimes n}$$

Measurement For classification, we measure the expectation values of Pauli-Z operators on each qubit:

$$\langle Z_i \rangle = \langle \psi_{\text{out}} | Z_i | \psi_{\text{out}} \rangle, \quad i \in \{0, 1, 2, 3, 4, 5, 6\}$$

While the circuit accepts an input parameter, our implementation uses random parameterization rather than direct encoding of input features. This approach allows us to explore the quantum feature space through stochastic sampling, which can be advantageous for certain classification tasks.

4.4 Quantum Convolutional Neural Network (QCNN)

The QCNN model consists of alternating quantum convolution and pooling layers followed by a variational layer for classification. Each convolutional unit applies a fixed entangling circuit to local qubit pairs, defined as:

$$U_{\text{conv}}^{(i,j)} = RZ_i(\theta) \cdot CX_{ij} \cdot RZ_j(\phi) \cdot CX_{ji} \cdot RZ_i(\lambda)$$

Pooling layers reduce the number of qubits to minimize data size, Quantum aggregation typically uses qubit measurements or control gates to discard unimportant information. A final variational block applies trainable rotations and entanglements to the reduced state, followed by measurement on selected qubits to obtain classification logits.

4.5 Hybrid QCNN + HMM Model

We propose a hybrid architecture that combines Quantum Convolutional Neural Networks (QCNN) with Hidden Markov Models (HMM). In this approach, we will not use QCNN as a standalone model, but we will use it to transform features extracted by standard feature extractors (e.g., MFCC, D-Vector) through a combination of quantum convolution, entanglement, and pooling operations as mentioned in Sections *Quantum Encoding and Circuit Design* and *Quantum Convolutional Neural Network (QCNN)*. We will then combine these transformed features with the original extracted features. The transformed QCNN features act as high-level representations, while the original features retain lower-level spectral and temporal cues. Then we will use the combined features to train the HMM for speaker identification.

Each hidden state in HMM represents a potential speaker identity, while the observation probabilities are computed using the enriched feature vectors. This combination allows the model to leverage both the discriminative power of quantum-enhanced feature transformation and the temporal consistency of HMMs, resulting in more robust speaker identification.

As mentioned in Fig 1, we will add an additional step after feature extraction, processing the extracted features through QCNN and then we will combine the original features with the transformed ones to train or identify speakers. Fig 2, show how it works.

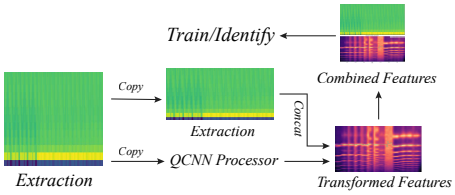

Fig. 2. QCNN Flow

4.6 Training and Evaluation

Our proposed models are trained using a cross-entropy loss on speaker labels, with parameter-shift rules applied to compute quantum gradients. For the hybrid model, once the QCNN is trained, we use the Baum-Welch algorithm to train the HMM based on the combined features (original + transformed). Evaluation is performed through cross-validation, and we report accuracy, precision, recall, and F1-score on both held-out speakers and unseen audio segments to assess generalization and robustness.

5 Experiments

5.1 Experiment Setup

We evaluated our proposed models using a curated Vietnamese speech dataset comprising various conversational contexts, including natural dialogues, group discussions, podcasts, lectures, and interviews. The total size of the dataset is 15GB, with many measurements listed below:

Table 1. Overview of speech dataset statistics

Statistic	**Value**
Total number of speakers	83
Total number of dialogues	10,430
Number of speaker changes	751
Total number of audio files	85
Total number of script files (.txt)	85
Total duration of dataset	64,608 seconds (17.93 hours)
Average dialogue length	6.19 seconds
Standard deviation of length	11.96 seconds

5.2 Evaluation Results

We use cross-validation to train and test many model–extractor combinations, and we obtained the results show in table 2.

Table 2. Evaluation Results of Different ASR Models Across Folds

Metric / Fold	HMM-MFCC	HMM-XVec	HMM-Wavelet	HMM-DVec	RNN-MFCC	RNN-XVec	RNN-Wavelet	RNN-DVec	QCNN-MFCC	QCNN-Wavelet	QCNN-XVec	QCNN-DVec	QCNN-HMM-MFCC	QCNN-HMM-Wavelet	QCNN-HMM-XVec	QCNN-HMM-DVec
Accuracy Fold 1	0.49	0.66	0.15	0.36	0.78	0.69	0.39	0.86	0.42	0.16	0.11	0.44	0.83	0.32	0.95	**0.97**
Accuracy Fold 2	0.55	0.71	0.18	0.36	0.79	0.70	0.39	0.86	0.42	0.21	0.11	0.49	0.82	0.33	0.95	**0.97**
Accuracy Fold 3	0.48	0.67	0.17	0.35	0.78	0.75	0.39	0.86	0.56	0.18	0.11	0.48	0.83	0.34	0.95	**0.97**
Precision Fold 1	0.74	0.67	0.23	0.04	0.59	0.27	0.15	0.80	0.06	0.04	0.00	0.07	0.89	0.26	0.96	**0.97**
Precision Fold 2	0.74	0.74	0.27	0.04	0.70	0.34	0.17	0.80	0.06	0.07	0.00	0.07	0.88	0.26	0.96	**0.97**
Precision Fold 3	0.76	0.66	0.22	0.04	0.63	0.39	0.17	0.80	0.36	0.04	0.00	0.08	0.88	0.27	0.97	**0.98**
Recall Fold 1	0.69	0.59	0.12	0.10	0.53	0.28	0.14	0.74	0.09	0.05	0.01	0.10	0.87	0.19	0.93	**0.95**
Recall Fold 2	0.73	0.62	0.15	0.10	0.58	0.32	0.15	0.73	0.07	0.06	0.01	0.13	0.85	0.19	0.90	**0.92**
Recall Fold 3	0.71	0.56	0.12	0.10	0.56	0.39	0.16	0.74	0.31	0.04	0.01	0.13	0.86	0.21	0.92	**0.94**
d F1-score Fold 1	0.64	0.59	0.11	0.05	0.54	0.25	0.13	0.75	0.07	0.03	0.00	0.07	0.87	0.18	0.94	**0.96**
F1-score Fold 2	0.66	0.63	0.14	0.05	0.59	0.30	0.14	0.74	0.05	0.04	0.00	0.08	0.85	0.17	0.92	**0.94**
F1-score Fold 3	0.66	0.57	0.13	0.05	0.57	0.37	0.15	0.75	0.30	0.03	0.00	0.09	0.86	0.19	0.94	**0.95**

5.3 Comparison of Models and Discussion

Table 2 presents the performance of different ASR models evaluated across three cross-validation folds. From the results, it is evident that there are significant differences in performance depending on the feature extraction methods and the model architectures used.

We experimented some traditional models like HMM combined with MFCC or Wavelet features achieved moderate accuracy, but their recall and F1-score remained low, indicating limited robustness in distinguishing between speakers. We then use pre-trained XVector and DVector features with HMM, that show a slight improvement in accuracy for XVector, but DVector alone performed poorly when used with HMM without any deep learning enhancement.

RNN-based models generally outperformed the pure HMM-based approaches. The RNN + DVector model achieved a very high accuracy (up to 86%) and consistent performance across folds, demonstrating the strength of the pretrained DVector embeddings in capturing speaker characteristics.

QCNN-based models alone (without HMM) showed inconsistent results, especially when combined with MFCC or XVector features. However, when QCNN was combined with HMM, there was a noticeable performance boost, particularly for models using DVector and XVector features. Among them, the **QCNN + HMM + DVector** model consistently achieved the highest performance across all folds, with an average accuracy of **97%**, precision of **97%**, recall of **94%**, and F1-score of **95%**.

This exceptional performance can be attributed to three key components:

- **DVector embeddings** provide highly discriminative speaker features extracted using deep neural networks trained for speaker verification, capturing robust speaker characteristics even in short utterances.
- **Quantum Convolutional Neural Networks (QCNN)** enhance the model's ability to learn non-linear and high-dimensional patterns from DVector features, which may not be fully captured by traditional CNNs.
- **Hidden Markov Models (HMM)** complement the QCNN by modeling temporal transitions in speech, which is effective in modeling and distinguishing speaker-specific vocal characteristics.

In summary, the combination of DVector embeddings, quantum-enhanced feature learning through QCNN, and temporal modeling via HMM offers a powerful and accurate architecture for speaker recognition tasks.

5.4 Limitations of QCNN Performance in Certain Configurations

While QCNN-based models show strong potential, their performance is highly dependent on the choice of input features and the architectural combination. As shown in Table 2, QCNN models combined with features like Wavelet or XVector often yield extremely low F1-scores (frequently below 0.2, and in some cases near zero), indicating that these configurations fail to capture meaningful patterns for speaker discrimination.

There are two key factors contributing to this instability:

- **Suboptimal feature selection**: Input features such as Wavelet and XVector, when not appropriately preprocessed or aligned with the model architecture, may lack sufficient speaker-specific information. In particular, handcrafted features or embeddings not tailored for the task can limit the learning capacity of the model.
- **Quantum model simulation limitations**: All QCNNs in this study are simulated on classical computers rather than executed on real quantum hardware. While simulation enables feasibility testing, it imposes severe computational constraints. As the number of qubits or circuit depth increases, training time grows significantly, and the risk of poor convergence also rises. These simulation-induced limitations may contribute to the poor and inconsistent performance of certain QCNN configurations.

In contrast, the combination of QCNN with HMM and DVector features demonstrates consistently superior results across all folds, with an average accuracy of **97%**, and F1-scores ranging from **0.92 to 0.98**. This indicates that the effectiveness of QCNN models is highly sensitive to both the feature representations and their integration with temporal models like HMM.

Overall, these findings emphasize that quantum-enhanced learning models like QCNN cannot be evaluated in isolation. Their performance is tightly coupled with the quality of input features and the computational feasibility of simulating quantum circuits. Careful design choices are required to fully leverage their advantages in practical ASR tasks.

6 Conclusion

This paper presents a comparative study of multiple speaker identification models, combining a range of feature extraction methods (MFCC, Wavelet, XVector, DVector) with machine learning techniques such as HMM, RNN, and QCNN. Among all the evaluated models, the QCNN + HMM + DVector combination achieved the best overall performance, with an average accuracy of 97% and an F1-score of 95% across three folds (see Table 2). These results highlight the strong synergy between deep speaker embeddings (DVector), quantum-inspired feature processing (QCNN), and sequential pattern modeling (HMM).

The main contributions of this work are as follows:

– We propose a hybrid speaker identification framework that leverages DVector-based embeddings, Quantum Convolutional Neural Networks (QCNN), and Hidden Markov Models (HMM).
– We provide a comparative analysis against a wide range of baseline systems, highlighting the strength of quantum-enhanced architectures.
– A custom dataset was constructed, and rigorous k-fold cross-validation was applied to ensure robust and fair evaluation.

The limitations of this work include:

– The quantum model training remains computationally intensive and may take a long training period.
– Evaluation was conducted under relatively clean and controlled audio conditions like conferences, tedtalks, podcasts, ..., without extensive testing in noisy or real-world environments.
– Generalization to unseen speakers from entirely different distributions was not assessed in this study.

Future work: We aim to explore more diverse datasets, including multiple languages and mixed-language speech. Our future work also focuses on improving the model's performance in noisy environments or when the audio quality is low. On the technical side, we plan to deploy the model in real-time applications such as voice assistants or security systems, potentially combining it with emotion detection for more advanced and context-aware interactions

In summary, our findings demonstrate the potential of integrating quantum-inspired learning with deep speaker representations for robust and accurate speaker identification, paving the way for more advanced speech-based technologies.

7 Acknowledgment

This research was supported by The VNUHCM-University of Information Technology's Scientific Research Support Fund.

References

1. Cong, I., Choi, S., Lukin, M.D.: Quantum convolutional neural networks. Nature Physics **15**, 1273–1278 (2019)
2. Davis, S.B., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Transactions on Acoustics, Speech, and Signal Processing **28**(4), 357–366 (1980)
3. Graves, A., rahman Mohamed, A., Hinton, G.: Speech recognition with deep recurrent neural networks. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) pp. 6645–6649 (2013)
4. Gupta, M.R., Gilbert, J.P.: Robust speech recognition using wavelet coefficient features. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). pp. I–613–I–616. IEEE (2003)

5. Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., Ng, A.Y.: Deep speech: Scaling up end-to-end speech recognition. arXiv preprint arXiv:1412.5567 (2014)
6. Ilyas, M.Z., Samad, S.A., Hussain, A., Ishak, K.A.: Speaker verification using vector quantization and hidden markov model. In: Proceedings of the 5th Student Conference on Research and Development (SCOReD). pp. 1–6. IEEE, Malaysia (2007). https://doi.org/10.1109/SCORED.2007.4451528
7. Perero-Codosero, J.M., Espinoza-Cuadros, F.M., Hernández-Gómez, L.A.: A comparison of hybrid and end-to-end asr systems for the iberspeech-rtve 2020 speech-to-text transcription challenge. Applied Sciences **12**(2), 903 (2022)
8. Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE **77**(2), 257–286 (1989)
9. Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., Khudanpur, S.: X-vectors: Robust dnn embeddings for speaker recognition. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). pp. 5329–5333. IEEE (2018)
10. Tufekci, Z., Gowdy, J.N.: Feature extraction using discrete wavelet transform for speech recognition. In: Proceedings of IEEE Southeastcon. pp. 116–123 (2000)
11. Variani, E., Lei, X., McDermott, E., Moreno, I.L., Gonzalez-Dominguez, J.: Deep neural networks for small footprint text-dependent speaker verification. In: 2014 IEEE international conference on acoustics, speech and signal processing (ICASSP). pp. 4052–4056. IEEE (2014)
12. Voll, K.: A hybrid approach to improving automatic speech recognition via nlp. In: Advances in Artificial Intelligence (Canadian AI 2007). pp. 514–525. Springer (2007)
13. Wan, L., Wei, X., Xie, L., Xu, Y.: D-vector: A deep learning approach for speaker verification. Interspeech **1600-1604**, 1600–1604 (2018)
14. Wan, L., Wang, Q., Papir, A., Moreno, I.L.: Generalized end-to-end loss for speaker verification. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). pp. 4879–4883. IEEE (2018)
15. Wang, K.C.: Robust voice activity detection based on discrete wavelet transform. In: Proceedings of the 20th Conference on Computational Linguistics and Speech Processing. pp. 216–228. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP), Taipei, Taiwan (2008)
16. Yang, C.H.H., Qi, J., Chen, S.Y.C., Chen, P.Y., Siniscalchi, S.M., Ma, X., Lee, C.H.: Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6523–6527. IEEE (2021)