RESEARCH ARTICLE

Adaptive Phoneme State Learning Architecture for Enhanced Speech Recognition Using Backpropagation Neural Network and Hidden Markov Model

[version 1; peer review: 1 approved, 2 not approved]

Rashmi Siddalingappa[1], Deepa S[2], Margaret Savitha[2], Kalpana P[2], Priya Stella Mary I[2], Shivanand Gornale[3], Lakshmi B A[4], Kefeng Li[5], Khang Wen Goh[6]

[1]Computer and Data Science, York St John University, London, England, E14 2BA, UK
[2]Christ University, Bengaluru, Karnataka, India
[3]Department of Computer Science, Rani Channamma University, Belagavi, Karnataka, India
[4]UST Global, Bangalore, Karnatake, India
[5]Macao Polytechnic University, Macau, Macao
[6]INTI International University & Colleges, Nilai, Negeri Sembilan, Malaysia

v1 **First published:** 02 Mar 2026, **15**:338
https://doi.org/10.12688/f1000research.177414.1
Latest published: 02 Mar 2026, **15**:338
https://doi.org/10.12688/f1000research.177414.1

Abstract

Speech remains a primary mode of human communication; however, automated speech recognition (ASR) systems face challenges from accent variability, temporal fluctuations, noise, and data privacy concerns. This paper proposes an enhanced ASR architecture incorporating an Adaptive Phoneme State Learning (APSL) algorithm with a Backpropagation Neural Network (BPNN) and Hidden Markov Model (HMM). APSL dynamically adjusts HMM state probabilities using phoneme confidence scores derived from the BPNN, thereby improving phoneme transition modeling and alignment. The multi-stage ASR pipeline includes noise reduction, speech-pause detection, and feature extraction via framing and windowing. APSL's adaptive mechanism reduces ambiguities in phoneme transitions, resulting in a more accurate speech-to-text conversion. A comparative evaluation framework assesses the baseline HMM, standalone BPNN, and integrated APSL-BPNN-HMM model. Experiments were conducted using a custom-built dataset of 2000 audio files alongside five benchmark corpora: BNC, ANC, COCA, Buckeye, and Emu. Key evaluation metrics—recall, precision, F-score, and Word Error Rate (WER)—demonstrate that the APSL-enhanced model significantly outperforms baseline systems, achieving 95.7% recall, 92.95%

Open Peer Review

Approval Status ✗ ✗ ✓

	1	2	3
version 1 02 Mar 2026	✗ view	✗ view	✓ view

1. **Hamza Kheddar**, University of Medea,, Medea, Algeria

2. **Fayzulla Nazarov**, Samarkand State University named after Sharof Rashidov (Ringgold ID: 187914), Samarkand, Uzbekistan

   Ilyos Khujayorov, Tashkent University of Information Technologies named after Muhammad al-Khwarizm (Ringgold ID: 187932), Tashkent, Uzbekistan

3. **Ramanda Rizky**, Universitas Lancang Kuning, Pekanbaru, Indonesia

Any reports and responses or comments on the

precision, 94.53% F-score, and 96% overall accuracy. Notably, APSL-BPNN-HMM consistently yielded the lowest WER across all datasets, validating its effectiveness. This work highlights the benefits of adaptive learning in probabilistic frameworks for achieving robust and accurate speech recognition.

Keywords
acoustic modeling, back propagation neural networks, hidden markov model, speech recognition, voice activity detection

article can be found at the end of the article.

Corresponding author: Rashmi Siddalingappa (r.siddalingappa@yorksj.ac.uk)

Author roles: Siddalingappa R: Conceptualization, Data Curation, Investigation, Methodology, Resources, Software, Validation, Visualization, Writing – Original Draft Preparation; **S D**: Formal Analysis, Validation, Writing – Review & Editing; **Savitha M**: Data Curation, Investigation, Resources, Visualization; **P K**: Conceptualization, Data Curation, Software, Validation; **Stella Mary I P**: Investigation, Methodology, Resources, Software; **Gornale S**: Project Administration, Supervision, Writing – Review & Editing; **B A L**: Data Curation, Resources, Validation; **Li K**: Supervision, Writing – Review & Editing; **Wen Goh K**: Investigation, Validation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: The author(s) declared that no grants were involved in supporting this work.

How to cite this article: Siddalingappa R, S D, Savitha M *et al.* **Adaptive Phoneme State Learning Architecture for Enhanced Speech Recognition Using Backpropagation Neural Network and Hidden Markov Model [version 1; peer review: 1 approved, 2 not approved]** F1000Research 2026, **15**:338 https://doi.org/10.12688/f1000research.177414.1

First published: 02 Mar 2026, **15**:338 https://doi.org/10.12688/f1000research.177414.1

1. Introduction

Speech is a dynamic cascade of thoughts produced by articulating utterances in natural language. The visual representation of language is called 'graphemes,' while the sound representation is called 'phonemes.' In linguistics, the study of phonemes encompasses "Phonetics" and "Phonology." Phonetics examines the physical properties of speech sounds, including their production by vocal organs (articulatory phonetics), auditory perception (auditory phonetics), and acoustic properties (acoustic phonetics). Phonology studies sound patterns and the systematic organization of sounds within a linguistic system.[1] These disciplines enable the transformation of graphemes into phonemes (text-to-speech, TTS) and vice versa (speech-to-text, STT). A speech recognition model (SRM) comprises three primary elements: i) Feature Extraction, which captures features and computes HMM states by transforming speech signals into spectral attributes mapped onto phonemic structures, yielding syllabic probability scores,[2] ii) Acoustic model, which identifies sound structures and extracts textual elements from spoken words,[3] and iii) Language model, which deciphers spectral attributes into meaningful word representations.[4] These processes require a pipeline architecture due to cross-language integration challenges. While training corpora must encompass all phoneme variations, storing every word-phoneme pair is impractical given memory and computational constraints. Machine learning addresses this through statistical models like HMM, enabling phoneme representation learning with limited data.[5] This research introduces the Adaptive Phoneme State Learning (APSL) algorithm, integrating a Backpropagation Neural Network (BPNN) with HMM to dynamically refine phoneme state transitions. The objectives are: i) develop a speech recognition interface for English phonemes, ii) transcribe spoken words into text, iii) enhance scalability and efficiency to reduce training time, iv) achieve human-level performance in real-time scenarios, and v) validate methodologies through comprehensive evaluation metrics including F-measure, recall, precision, and accuracy. The paper is structured as follows: Section 2 reviews HMM-based speech recognition literature, Section 3 outlines the architectural model and methodology, Section 4 explains voice activity detection and textual computation algorithms, Section 5 discusses the experimental setup, Section 6 presents results and future directions, and Section 7 concludes the study.

2. Research background

The roots of phonetics trace back to as early as 500 BC on the Indian subcontinent, with Panini meticulously describing the place and manner of articulation of consonants in Sanskrit.[6] The chronicles of speech recognition date to 2002, culminating in a final output release in 2005, functioning proficiently across three languages: English, Spanish, and Mandarin.[7] Operating at a speech rate of 10 Hz with a recording precision of 96 kHz/24 bit, this innovation marked a pivotal milestone. Fast-forward to 2019, another speech synthesizer emerged during the "Blizzard challenge",[8] pronouncing 1200 phonetic utterances at a frequency of 1.5 Hz. Several researchers have contributed to the advancement of HMM-based speech recognition systems, as summarized in Table 1. These studies demonstrate various approaches to phonetic segmentation, speech synthesis, and recognition across different languages and acoustic conditions. While these prior works have made significant contributions, they exhibit certain limitations including moderate accuracy levels, language-specific implementations, and challenges in handling diverse speech qualities. Against this backdrop, the present study introduces several key innovations: 1) labeling synthetic waveforms with distinct features, 2) employing MFCC filtering to dynamically extract feature coefficients as an energy measure, 3) addressing the challenge of insufficient training observations in HMM models by encompassing both forward and backward training spectral features. This innovation also introduces time-dependent windowing factors to reduce memory requirements and optimize likelihood summation across all states, thereby elevating accuracy, and 4) the proposed model demonstrated remarkable accuracy even in noisy environments.

3. Architecture of speech recognition model for speech-to-text process

The proposed APSL-BPNN-HMM architecture integrates multiple components to enhance speech recognition through effective signal processing and machine learning, as shown in Figure 1. The input audio signal is processed through a Speech Acquisition module for proper sampling and data segmentation. Given the stochastic nature of speech signals, Voice Activity Detection (VAD) distinguishes between speech and non-speech regions, improving noise reduction and signal normalization. Feature Extraction employs Mel-frequency cepstral coefficients (MFCC) with preprocessing steps including pre-emphasis (boosting high frequencies) and framing (segmenting data into manageable frames), retaining essential phonetic and linguistic information. The extracted features undergo windowing, segmenting frames into overlapping windows activated using bi-gram lexicon combinations to ensure meaningful word boundaries. The Adaptive Piecewise Segment Labeling (APSL) module enhances segment identification and labeling, improving feature sequence reliability for model training. The labeled features are fed into a Backpropagation Neural Network (BPNN), which refines feature representations and generates intermediate outputs for the Hidden Markov Model (HMM).[17] The HMM models temporal dependencies and stochastic patterns, segmenting speech into phonemes, words, and sentences. Bi-gram connections model phoneme and word transitions, ensuring improved accuracy. The speech recognition module identifies and classifies predicted speech patterns, with performance evaluated using Accuracy, Precision, Recall, F1-score, and Word Error Rate (WER). This architecture effectively addresses noise reduction, signal normalization,

Table 1. Literature survey summary.

Refs.	Problem/Focus	Core method	Datasets/Setup	Key findings	Limitations
[9]	Homophonic ambiguities in Malay name retrieval	Soundex and Asoundex methods for generating name codes	Malay names corpus	Improved accuracy by 38.3% compared to prior methods	Limited to name retrieval; not applicable to continuous speech recognition
[10]	Cross-language phonetic segmentation	HMM-based phonetic segmentation framework	Appen Spanish speech corpus	Achieved approximately 61.5% accuracy	Moderate accuracy; requires improvement for practical deployment
[11]	Phonetic-based recognition of semivowel sounds	Comparison of HMM and MFCC-based recognizers	T146 database	Explored novel avenues in phonetic analysis of semivowels	Specific to semivowel recognition; limited generalization to broader phoneme classes
[12]	Phonetic segmentation based on speech analysis	Microcanonical Multiscale Formalism (MMF) technique	Speech corpus with varied phonetic contexts	6% improvement in segmentation accuracy	Modest accuracy gains; computational complexity not addressed
[13]	Arabic speech recognition with pronunciation variations	HMM for associating diverse pronunciations	Arabic speech corpus	Minimized phonetic out-of-vocabulary rate; demonstrated HMM efficacy	Language-specific; limited discussion of cross-linguistic applicability
[14]	Speech synthesis for Indian English syllables	HMM-based speech synthesizer	Indian English syllable dataset	Achieved 89% accuracy	Syllable-word model not delineated; accuracy limited for complex utterances
[15]	Murmured speech recognition and conversion	HMM with posterior decoding approach	Murmured speech dataset	Attained 81.2% accuracy in murmur-to-normal speech conversion	Moderate accuracy; challenges in handling diverse speech qualities
[16]	Speech recognition using time and frequency analysis	HMM with time and frequency response extraction techniques	Standard speech corpus	Explored feature extraction methods for HMM-based recognition	Limited performance metrics reported; scalability not discussed

F1000Research 2026, 15:338 Last updated: 06 MAY 2026

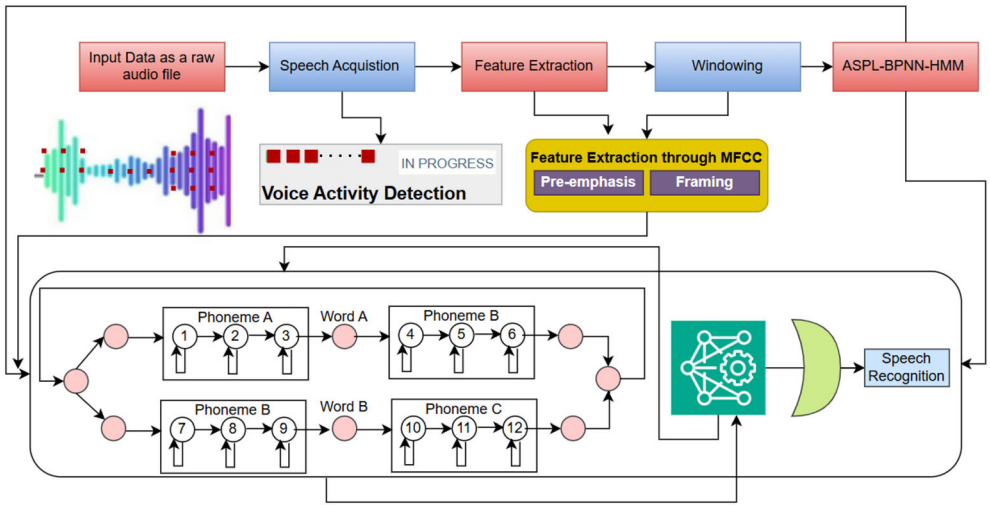


Figure 1. APSL-BPNN-HMM Architecture for Speech Recognition — The proposed architecture integrates key components such as Voice Activity Detection (VAD), Mel-frequency cepstral coefficients (MFCC) based feature extraction with pre-emphasis and framing, Adaptive Piecewise Segment Labeling (APSL) for enhanced segmentation, and a combination of Back Propagation Neural Network (BPNN) and Hidden Markov Model (HMM).

and robust speech recognition in dynamic environments through integrated APSL segmentation, MFCC-based feature extraction, and HMM temporal modeling.

3.1 Speech acquisition

Raw speech signals are acquired through microphones, online audio files, or audio CDs. Accurate sampling frequency configuration is critical before recording. For example, a 100-second audio file sampled at 44100Hz yields $44100 \times 100 = 4{,}410{,}000$ samples, ensuring CD-quality audio. Based on Nyquist's theory,[18] the sampling rate must be at least twice the maximum signal frequency to avoid aliasing. For instance, a 10,000 Hz signal requires a minimum 20,000 Hz sampling rate. Sampling frequency selection involves a trade-off between audio quality and memory consumption: lower frequencies reduce memory usage but compromise quality, while higher frequencies enhance fidelity at the cost of increased storage. The optimal balance depends on application-specific requirements.

3.2 Voice Activity Detection (VAD)

Voice Activity Detection (VAD) comprises two stages: noise removal and speech pause detection. Noise elimination employs a Training-Based Noise Removal Technique (TBNRT),[19] utilizing a corpus of noise types from white to environmental noise. Noise segments matching the noise dictionary are removed using high-pass and low-pass filters. Endpoint detection utilizes algorithms based on energy variance, pitch modulation, zero-crossing rate, cepstral parameters, or linear prediction coding (LPC).[20] VAD applies the min/max energy threshold (ET) paradigm. For sample S_{B_i} in each speech segment B_i, ET is defined at indices x and y, where x represents the total signal duration and y represents the duration within block B_i. S_i denotes the speech signal in each segment, where $S = \{1, 2, \ldots, n\}$.

Step–1: The energy is calculated using Equation (1):

$$E_x(x) = \sum_{y \in S_{B_i}}^{N} S_{f_i}^2(x) \tag{1}$$

Step–2: Voice Activity Detection (VAD - Equation 2)

$$B_x(x) = \begin{cases} 1, & T_M(x) \geq T_B \\ 0, & T_M(x) < T_B \end{cases} \tag{2}$$

where T_m and T_M are the minimum and maximum thresholds, respectively, and T_B is the base threshold.

Step–3: When T_m is reached, the signal breaks until the next T_m is reached.

VAD extracts speech features every 5-40 ms and compares them to base threshold T_B. Features exceeding T_B yield VAD = 1 (speech present); otherwise VAD = 0 (no speech). Initially assuming a 40 ms segment contains no speech, we analyze frames of 60 samples (6 ms duration) collected at 70 kHz. The average threshold for each frame is determined using Equation (3):

$$T_{\text{mean}} = \frac{1}{M}\sum_{n=0}^{N} T_x \tag{3}$$

Since loudness varies among speakers, we focus on minimum loudness. Using Praat,[21] we analyzed loudness ranges to categorize T_m, employing a Python script to eliminate signals at the T_m threshold. For instance, the quietest sound measured 59.3 dB, with quiet segments ranging from 59-62 dB. The first segment below T_B is designated as T_m. Speech typically begins softly, peaks at maximum T_M, then decreases, defining the minimum-maximum energy range. The quiet threshold is set at -25.0 dB, with segments below classified as quiet. Temporal constraints include a minimum pause duration of 0.1 seconds between words (longer for sudden loud sounds; shorter durations are not classified as quiet) and a minimum sounding time of 0.05 seconds (representing inter-syllable pauses).

3.3 Feature extraction

Feature extraction techniques include mel-frequency cepstral coefficients (MFCC),[22] vector quantization (VQ),[23] artificial neural networks (ANN),[24] Hidden Markov Models (HMM),[25] and dynamic time warping (DTW).[26] This study employs MFCC for framing and HMM for windowing. MFCC-based feature extraction involves two steps: Pre-emphasis and Framing.

Pre-emphasis: High-frequency sounds typically have lower magnitudes, leading to higher distortion and compromised speech quality. Pre-emphasis counters this by suppressing high-frequency components and boosting magnitude, producing a smoother profile than the original audio. The pre-emphasis factor α is calculated using Equation (4):

$$\alpha = \exp\left(-2\pi\nu T/\lambda c\right) \tag{4}$$

where f represents the audio signal frequency and T represents the sampling period. For each sample except the first, the alteration follows Equation (5):

$$X_k = X_k - \alpha X_{k-1} \tag{5}$$

Framing: Framing is a lossless process that divides continuous signals into overlapping, time-specific frames to reduce transition discontinuities. Using MFCC filtering, sound samples are represented as time functions with coefficients for frames centered at equally spaced intervals. Each speech segment—sounding or silent—is treated as a frame, with total frames equal to the sum of utterances and pauses. For example, the sentence “The joy of living is to love and respect” (5.871 s) includes utterances: “the” = 0.14 s, “joy” = 0.39 s, “of” = 0.08 s, “living” = 0.56 s, “is” = 0.10 s, “to” = 0.12 s, “love” = 0.47 s, “and” = 0.24 s, “respect” = 0.72 s, and pauses: 1.08, 0.26, 0.14, 0.33, 0.16, 1.06 s. The sounding (2.823 s) and silent durations (3.043 s) sum to the total (5.871 s), ensuring accurate, lossless framing.

4. Materials and methods

4.1 Data

Broad representativeness requires a sufficiently large training dataset including utterances from male and female speakers. Since speech varies significantly across phonetic contexts, a comprehensive model requires at least 100,000 sentences. Manual recording is highly labor-intensive, involving content selection, phonetic variation coverage, participant recruitment, post-processing, and transcription. We utilized publicly available speech corpora, including the British National Corpus (BNC),[27] American National Corpus (ANC),[28] and Corpus of Contemporary American English (COCA),[29] selecting the Buckeye Speech Corpus[30] and EMU Speech Database[31] for training. Buckeye comprises approximately 40 hours of conversational English (360,000 words or 24,000 sentences at 15 words/sentence). EMU contributes 30,000 sentences, yielding 54,000 total sentences. To meet the desired data volume, we applied augmentation techniques including pitch shifting (adjusting pitch without affecting duration to simulate various speaker profiles), time-stretching (modifying speech speed while preserving pitch for different speaking rates), volume alteration, background noise addition, and reverberation simulation to introduce acoustic variability. These methods increased the

effective dataset to approximately 150,000 sentences. For storage, assuming mono audio at 16 kHz sampling rate and 16-bit resolution (32 KB/second), with 150,000 sentences averaging 5 seconds each as described in Equation 6:

$$\text{Storage} = 150{,}000 \times 5 \text{ sec} \times 32 \text{ KB/sec} = 24{,}000{,}000 \text{ KB} \approx 24 \text{ GB} \tag{6}$$

The model is evaluated on all five corpora. Speech recognition tasks were implemented using *Praat,*[21] a phonetic analysis tool developed by Paul Boersma and David Weenink at the Amsterdam Institute of Phonetic Sciences, facilitating analysis, synthesis, and manipulation of speech signals for phonetics research.

4.2 Windowing through Hidden Markov model

Each speech signal frame captures cepstral features characterizing the corresponding sound segment. Windowing derives grapheme-level representations for each phoneme within a frame. Hidden Markov Models (HMMs) generate sequences and patterns of hidden states based on observed acoustic features, facilitating phoneme-to-grapheme mapping. During preprocessing, speech signal S is segmented into frames $\{f_n\}$, with each frame f_i subdivided into windows $\{w_n\}$, where each window w_i spans 0.015 s—optimal for preserving spectral information without temporal overlap or resolution loss. This is defined in Equations (7) and (8):

$$S := \{G_1, \ldots, G_k\} f_k := \{w_1, \ldots, w_k\} \tag{7}$$

$$S := \sum_{k=1}^{\infty} f_k \left(\sum_{s=1}^{\infty} w_s \right) : \forall w | \ll 0.001 \text{ sec} \tag{8}$$

Window formation follows Algorithm 1. Each acoustic feature extracted from a window maps to its corresponding language model component. Training the HMM classifier is crucial for accurate phoneme extraction. During training, known state sequences enable inference of unknown states. Training corpora include sound utterances for all syllable combinations with corresponding phonemic representations. Temporal overlap between consecutive windows or frames captures transitional features from previous states, improving current state learning. The overlap must balance containing at least one complete phoneme structure while avoiding excessive repetition. Based on empirical evaluation, overlap duration was set to 0.5 milliseconds between successive windows and frames.

Algorithm 1: HMM-based Windowing Process

Algorithm 1. HMM-based Windowing Process.

1: **Input:** Frames: f_n

2: **Output:** Each frame f_i was further divided into windows w_n with a length of 0.015 s.

3: **for** each frame f_i **do**

4: **for** each word X_i **do**

5: Compute the length of X_i, denoted as L_i.

6: Divide L_i by l_i, where $l_i = 0.015$ sec.

7: Consider the fractional part as the number of complete windows and the real part as the last window with adjusted length.

8: Count the number of complete windows, denoted as W_T.

9: Compute the total sum of window lengths:

$$S_T = \sum_{n=1}^{W_T} W_T(0.015) \tag{9}$$

10: Compute the length of the last window:

$$L_n = L_i - S_T \tag{10}$$

11: **end for**

12: **end for**

Language features are extracted from each window as follows. For every window (w_i), the corresponding phoneme is identified by matching acoustic features with pronunciation dictionary entries. If a unique phoneme is found, it is directly assigned and the process continues. When multiple phoneme candidates exist, probabilities are computed based on previously known state sequences, selecting the most probable phoneme. If no match is identified, an HMM infers the current state from prior known states. Finally, a dynamic text wrapping algorithm structures the phoneme combinations derived through HMM.

4.3 Backpropagation Neural Network (BPNN) in speech recognition

BPNN minimizes classification errors in speech-to-text conversion.[32] Feature extraction techniques such as mel-frequency cepstral coefficients (MFCCs) transform raw audio into feature vector $\mathbf{x}$, which BPNN processes to classify phonemes. Forward propagation computes neuron outputs in hidden and output layers:

$$a_j = f\left(\sum_{i=1}^{n} w_{ij} x_i + b_j\right), \tag{11}$$

where w_{ij} represents the weight between the i-th input neuron and j-th hidden neuron, b_j is the bias term, and $f(\cdot)$ is the activation function (sigmoid or ReLU):

$$f(z) = \frac{1}{1+e^{-z}} \quad \text{or} \quad f(z) = \max(0, z). \tag{12}$$

The output layer generates predicted phoneme probability distributions, with error calculated using cross-entropy loss:

$$L = -\sum_{k=1}^{m} y_k \log \hat{y}_k, \tag{13}$$

where y_k is the actual phoneme label and $\hat{y}_k$ is the predicted probability.

During backpropagation, error gradients are computed and propagated backward to adjust weights following gradient descent:

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} - \eta \frac{\partial L}{\partial w_{ij}}, \tag{14}$$

where η is the learning rate. Gradients are computed using the chain rule:

$$\frac{\partial L}{\partial w_{ij}} = \delta_j a_i, \tag{15}$$

where δ_j is the error term at neuron j.

4.4 Algorithm: Adaptive Phoneme State Learning (APSL)

This algorithm enhances traditional BPNN-HMM speech recognition by introducing an adaptive mechanism that refines HMM state transitions based on BPNN confidence scores. The Adaptive Phoneme State Learning (APSL) algorithm combines a BPNN and HMM to dynamically learn phoneme transitions. Speech signals are segmented into overlapping 0.015 s windows, with cepstral features extracted using MFCCs. The Viterbi algorithm identifies the most probable phoneme state sequence by maximizing transition likelihoods given the trained HMM parameters,[33] while the BPNN classifies phonemes and updates weights via gradient descent using the cross-entropy loss function.

$$L = -\sum_{k=1}^{m} y_k \log \hat{y}_k \tag{16}$$

where y_k is the true phoneme label, and $\hat{y}_k$ is the predicted probability.

The confidence score in the APSL represents the reliability of phoneme classification using BPNN. This is defined as the posterior probability $P(p_j|x)$, where p_j is a phoneme and x is the feature vector.[34] The confidence score helps in adaptive transition refinement, ensuring that phonemes with low classification certainty undergo additional training or an extended analysis.

If a phoneme's confidence score is below a threshold θ, APSL dynamically modifies the HMM transition and emission probabilities. The updated emission probability is computed as:

$$P(w_i|q_t) = \alpha P(p_j|x) + (1-\alpha)P_{HMM}(w_i|q_t) \tag{17}$$

where α is a weighting factor that balances the neural network output with the traditional HMM probability estimates.

To further improve recognition, APSL dynamically adjusts the window size for phonemes with low confidence scores:

$$w_i' = w_i + \Delta t, \quad \Delta t = 5ms \tag{18}$$

where w_i' is the updated window length.

The final phoneme sequence is determined by the Viterbi decoding process:

$$Q^* = \arg\max_Q P(Q|W) \tag{19}$$

where $W = \{w_1, w_2, \ldots, w_N\}$ represents the sequence of analyzed windows. The APSL model adapts over time by adjusting state transitions based on the observed confidence scores, reducing phoneme classification errors, and improving speech recognition accuracy.

Algorithm 2. Adaptive Phoneme State Learning (APSL) using BPNN-HMM.

Input: Speech signal S, predefined phoneme set P, HMM states Q
Output: Optimized phoneme sequence Q^*
Step 1: Preprocessing and Feature Extraction
Convert speech signal S into frames f_n with 15ms windows w_i
Extract Mel-Frequency Cepstral Coefficients (MFCCs) to form feature vectors x
Step 2: BPNN-Based Phoneme Probability Estimation
Train a BPNN model to classify phonemes
Compute phoneme confidence score $P(p_j|x)$ for each phoneme p_j
Step 3: Adaptive HMM Transition Refinement
for each state $q_t \in Q$ **do**
    Compute modified emission probability:
    $P(w_i|q_t) = \alpha P(p_j|x) + (1-\alpha)P_{HMM}(w_i|q_t)$
end for
Step 4: Dynamic Windowing for Phoneme Alignment
if $P(p_j|x) < \theta$ (confidence threshold) **then**
    Extend window: $w_i' = w_i + \Delta t, \quad \Delta t = 5ms$
end if
Step 5: Decoding with APSL
Apply Viterbi algorithm to obtain optimal phoneme sequence:
$Q^* = \arg\max_Q P(Q|W)$ where $W = \{w_1, w_2, \ldots, w_N\}$
Step 6: Training Updates using Backpropagation
Compute loss: $L = -\sum_{k=1}^{m} y_k \log \hat{y}_k$
Update BPNN weights:
$w_{ij}^{(t+1)} = w_{ij}^{(t)} - \eta \frac{\partial L}{\partial w_{ij}}$
Return Optimized phoneme sequence Q^*

4.4.1 Optimal hyperparameter tuning using Bayesian optimization

Hyperparameter tuning is a critical step in machine learning for identifying the optimal set of hyperparameters to enhance model performance. Unlike model parameters learned during training, hyperparameters are predefined and govern the

learning process, including the learning rate, number of hidden layers, batch size, and dropout rate. Selecting appropriate hyperparameters is essential for maximizing accuracy and minimizing errors. Bayesian Optimization is an efficient method for hyperparameter tuning, especially for complex models with expensive evaluation costs.[35] It constructs a probabilistic model of the objective function and uses an acquisition function to balance exploration and exploitation when selecting new hyperparameter configurations. Using Bayesian Optimization, optimal hyperparameters were determined for both the BPNN and APSL-BPNN-HMM speech recognition models. For the BPNN model, the optimal learning rate was 0.005, with three hidden layers of 256 neurons each, a batch size of 64, and 150 training epochs. The model employed the ReLU activation function with a dropout rate of 0.3, along with the Adam optimizer and cross-entropy loss function. For the APSL-BPNN-HMM model, the optimal learning rate was 0.003, with two hidden layers of 128 neurons each, a batch size of 64, and 200 epochs. The ReLU activation function with a dropout rate of 0.4 was used, while the confidence threshold (θ) was set to 0.75, the weighting factor (α) to 0.5, and the dynamic window adjustment size (Δt) to 15 ms. The Adam optimizer and cross-entropy loss function were also applied to ensure stable convergence and improved speech recognition accuracy.

4.5 An illustrated example

4.5.1 Frequency and probability calculations using HMM approach

Here, frequency indicates the number of times the corpus encounters the syllable. The probability of an individual syllable is obtained by dividing it by the total number of words in the corpus containing that syllable. ω represents any sequence of phonemes. Note: Only 2 words are shown, and the same process is repeated for other words in the given context.

The

Frequency = 87

$$P(t|0,0) = \text{Probability of 't' coming first} = \frac{31}{87} = 0.35$$

$$P(h|t,0) = \text{Probability of 'h' coming after 't' at the beginning} = \frac{19}{87} = 0.21$$

$$P(e|h,t) = \text{Probability of 'e' coming after 'th'} = \frac{29}{87} = 0.33$$

Therefore, each phoneme is now transformed into its corresponding syllable, 'the' → ðə,ðɪ,ðiː/

Using the pronunciation of 'the' as trained data, more words containing 'the' sequence such as this, there, these, then, and thesis are tested. These words are correctly recognized and converted to the exact match of a syllable.

Joy

Frequency = 14

(Joy was rejected, words in the dictionary are: jinx, job, jockey, jury, subject, disjoint, jealous, injury, rejoice, adjective, adjourn, rejected, conjure)

$$P(j|0,0) = \text{Probability of 'j' coming first} = \frac{5}{14} = 0.38$$

$$P(o|j,0) = \text{Probability of 'o' coming after 'j' at the beginning} = \frac{2}{14} = 0.15$$

$$P(y|o,j) = \text{Probability of 'y' coming after 'jo'} = 0$$

"joy" pattern was not found in the speech corpus. Therefore, with the help of HMM, the given phonemes are split into 2 different probabilities as follows:

1) 'jo' → $P(o|j,\omega)$, probability of 'o' coming after 'j', that is, $\frac{2}{14}$ + any other words in the dictionary withthe simple combination of 'jo'.of 'jo'. The words rejoice and adjourn are found in the dictionary, suiting this criterion. Thus, the total probability will be

$$\frac{2+2}{14} = 0.26$$

2) 'oy' $\rightarrow P(y|o,\omega)$. When searched in the corpus, the phoneme for 'oy' was found in the word 'annoy' pronunciation. Thus, the probability will be

$$\frac{1}{14} = 0.07$$

Supposedly, if 'jo ω' was not found and 'ω oy' was not found, then the HMM model will look for:

- 'ω j ω' alone (James) - 'ω o ω' (of) - 'ω y ω' (why)

Therefore, each phoneme of the word 'joy' $\rightarrow$ dʒ ɔɪ/ is now transformed into its corresponding syllable. ω represents any sequence of phonemes.

4.5.2 APSL-BPNN-HMM refinements, where phoneme probabilities are adjusted using BPNN confidence scores.

Here, frequency indicates the number of times the corpus encounters the syllable. The probability of an individual syllable is obtained by dividing it by the total number of words in the corpus containing that syllable. With APSL, the probability calculations are adjusted dynamically using BPNN-generated phoneme confidence scores. Let ω represent any sequence of phonemes.

The

Frequency = 87

$$P(t|0,0) = \text{Probability of } \text{`t'} \text{ coming first} = \frac{31}{87} = 0.35$$

$$P(h|t,0) = \text{Probability of } \text{`h'} \text{ coming after } \text{`t'} \text{ at the beginning} = \frac{19}{87} = 0.21$$

$$P(e|h,t) = \text{Probability of } \text{`e'} \text{ coming after } \text{`th'} = \frac{29}{87} = 0.33$$

With APSL-BPNN-HMM, each probability is updated with the BPNN confidence score (C) for each phoneme transition:

$$P'(e|h,t) = P(e|h,t) \times C(e)$$

If $C(e) = 0.95$, the adjusted probability is:

$$P'(e|h,t) = 0.33 \times 0.95 = 0.31$$

Thus, each phoneme is now transformed into its corresponding syllable:

'the' $\rightarrow$ ðə,ðɪ,ðiː/

With APSL-BPNN-HMM, phoneme sequences for words like this, there, these, then, and thesis are dynamically re-evaluated, leading to improved recognition accuracy.

Joy (Previously Rejected)

Frequency = 14

Previous HMM-based probabilities:

$$P(j|0,0) = \frac{5}{14} = 0.38$$

$$P(o|j,0) = \frac{2}{14} = 0.15$$

$$P(y|o,j) = 0$$

APSL Adjustment Using BPNN Confidence (C):

- BPNN assigns confidence scores based on phoneme similarity.
- Let $C(o) = 0.85$ and $C(y) = 0.78$.

Updated probability calculations:

$$P'(o|j,0) = P(o|j,0) \times C(o) = 0.15 \times 0.85 = 0.127$$
$$P'(y|o,j) = P(y|o,j) + (C(y) \times 0.1) = 0 + (0.78 \times 0.1) = 0.078$$

Now, **'joy'** is re-evaluated under APSL-BPNN-HMM and no longer rejected, as confidence-adjusted probabilities allow for better phoneme transition predictions.

4.5.3 Dynamic text wrapping

Dynamic text wrapping is applied each time phonemes are mapped between windows, wrapping and merging words after HMM processing. When acoustic features are involved, this is called dynamic time warping. Consider n lexical pairs formed in each window (w_i) for frame (f_i). Feature duplication occurs between previous (w_{n-1}) and present (w_n) windows due to the 0.5 ms overlap region. The process:

- Compare the last alphabet of the previous window ($w_{n-1}(a_n)$) with the first alphabet of the present window ($w_n(a_1)$).
- If identical, delete one and concatenate the remaining alphabets.
- Repeat for all windows (w) across all frames (f_n).

Here, a denotes an alphabet, with subscripts indicating position within a word. According to the HMM model, the phoneme "the" segments as follows:

- Window 1 (W_1) compared with window 2 (W_2):

$$W_1(a_n) \sim W_2(a_1), \quad W_1(t) \sim W_2(t)$$

Since 't' appears in both, cancel one 't'. Remaining: "t".

- Window 2 (W_2) compared with window 3 (W_3):

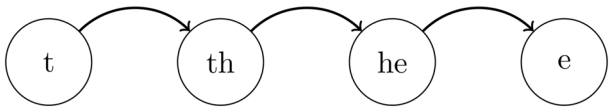
$$W_2(a_n) \sim W_3(a_1), \quad W_2(h) \sim W_3(h)$$

Since 'h' appears in both, cancel one 'h'. Remaining: "th".

- Window 3 (W_3) compared with window 4 (W_4):

$$W_3(a_n) \sim W_4(a_1), \quad W_3(h) \sim W_4(e)$$

Since $h \neq e$, keep 'e'. Final sequence: "the".



Memory Efficiency: Dynamic text wrapping links acoustic features with language parameters without requiring memory storage. An array stores frame contents where: i) array size is determined by the number of frames, ii) memory

addresses are allocated in ascending order as words form, and iii) wrapped texts are stored efficiently. At completion, words are concatenated as follows (refer to Table 2):

Table 2. Phoneme dynamic wrapping table for the example sentence.

The	Joy	Of	Living	Is	To	Love	And	Respect
A0	A1	A2	A3	A4	A5	A6	A7	A8

5. Results and discussions

5.1 Experimental set-up

The preprocessing phase is crucial for accurate and efficient speech recognition. To establish a robust dataset, 1000 audio files were manually created using mono channel setup with participants spanning ages 15-80, including fluent and non-fluent English speakers. All participants provided informed verbal consent following institutional ethical guidelines, as the study posed minimal risk and involved no sensitive personal data. Each participant used microphones and was presented with varying-length sentences. To introduce real-world variability, recordings were deliberately subjected to white and environmental noise. Noise was subsequently removed using high-pass and low-pass filters based on the Tunable Band Noise Reduction Technique (TBNRT) described in Section 3.2. The high-pass filter suppresses low-frequency noise:

$$H_{hp}(f) = \frac{f}{f_c} \quad \text{for} \quad f > f_c \tag{20}$$

The low-pass filter attenuates high-frequency noise:

$$H_{lp}(f) = \frac{f_c}{f} \quad \text{for} \quad f < f_c \tag{21}$$

where f_c is the cutoff frequency based on detected noise profiles.

Recordings were conducted at four sampling frequencies: 18000, 32300, 44100, and 56000 Hz. Empirical results demonstrated superior performance at 44100 Hz, providing optimal balance between memory efficiency and audio clarity. Consequently, 44100 Hz was designated as the standardized sampling frequency. Additionally, 1000 audio files were sourced from online platforms featuring male and female speakers with diverse accents, including English and non-English speakers. The dataset covers multiple regions: i) Western European, ii) Eastern European, iii) Central Asia/Middle East/North African, iv) Sub-Saharan Africa, v) South Asia, vi) South East Asia, vii) CJK (Chinese, Japanese, Korean).[36] This expanded dataset totals 2000 files (3 seconds to 3 minutes duration, 0.9 GB storage), plus 24 GB from the corpus detailed in Section 4.1, posing substantial memory challenges during training. To address memory overhead, the APSL-BPNN-HMM framework employs an Adaptive Phoneme State Learning (APSL) mechanism for efficient parameter utilization. APSL introduces adaptive parameter sharing, dynamically assigning model parameters across layers to reduce redundancy through shared weight matrices between neighboring phoneme states. Consider a BPNN layer with n input neurons, m hidden neurons, and p output neurons. Without APSL, total parameters are:

$$\Theta = (n \times m) + (m \times p) + b \tag{22}$$

where b represents bias terms. APSL defines shared parameter matrices W_s for phoneme states with similar acoustic properties, reducing independent parameters:

$$\Theta' = (n \times k) + (k \times p) + b \tag{23}$$

where $k < m$ represents the reduced dimensional space through adaptive sharing. APSL dynamically adjusts k based on phoneme similarity, reducing complexity without compromising accuracy.

APSL integrates dynamic thresholding for parameter sharing control. During training, a similarity matrix S_{ij} is computed between phoneme states i and j:

$$S_{ij} = \frac{\sum_{t=1}^{T} \phi_i(t) \cdot \phi_j(t)}{\sqrt{\sum_{t=1}^{T} \phi_i^2(t) \sum_{t=1}^{T} \phi_j^2(t)}} \tag{24}$$

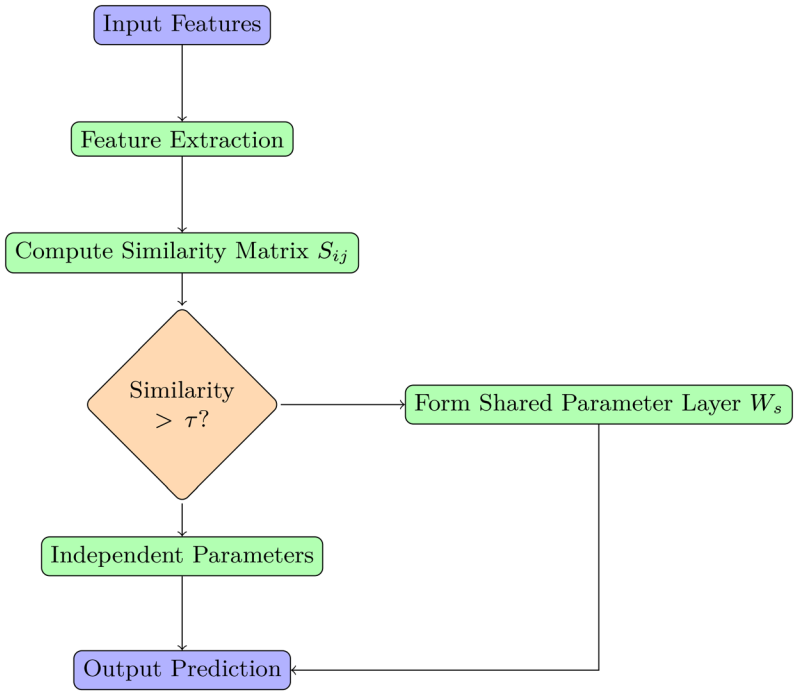


Figure 2. A flowchart depicting the APSL mechanism from input features through feature extraction, similarity matrix computation, and threshold-based decision making to form shared or independent parameters, culminating in the final prediction.

where $\phi_i(t)$ and $\phi_j(t)$ are feature vectors of phoneme states i and j at time t. If S_{ij} exceeds threshold τ, phoneme states are grouped under a shared parameter layer (see Figure 2). This adaptive parameter sharing significantly reduces redundant storage, optimizing memory usage from 24 GB to approximately 15.12 GB. This reduction mitigates hardware constraints and accelerates model convergence by limiting parameter explosion, ensuring efficient resource utilization and scalability for large-scale speech recognition tasks.

Figure 3 demonstrates the optimization impact by comparing memory consumption across iterations for the baseline and proposed APSL-BPNN-HMM framework. The Baseline Model (skyblue) steadily increases memory usage, reaching

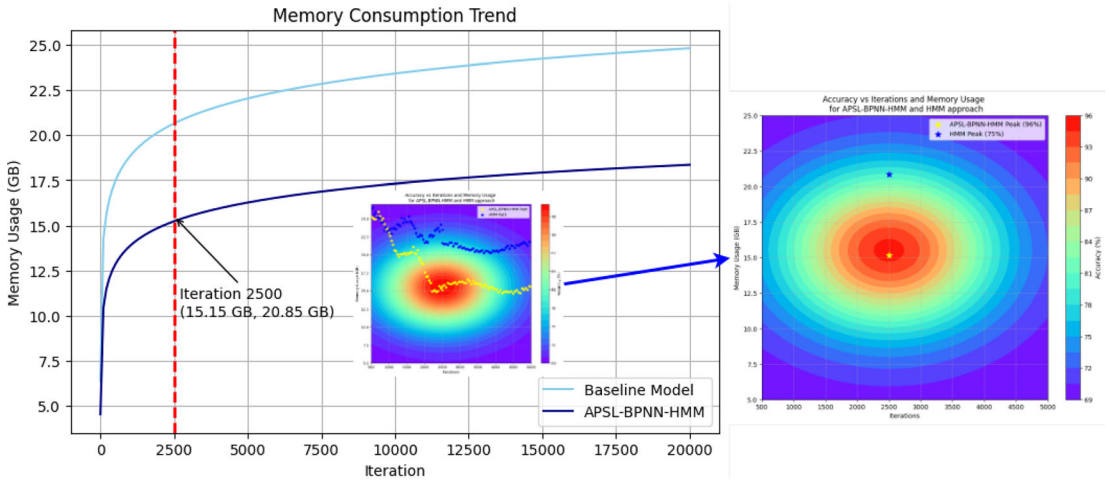


Figure 3. A line graph compares memory consumption across training iterations for the Baseline Model and APSL-BPNN-HMM, demonstrating reduced memory usage with adaptive parameter sharing, while an inset plot shows accuracy fluctuations for both models and an extended contour visualization highlights the memory–accuracy tradeoff at peak performance points.

approximately 20.85 GB at 5000 iterations, while APSL-BPNN-HMM (navy) maintains significantly lower usage, stabilizing around 15.15 GB. This reduction reflects APSL's effectiveness in minimizing redundant parameter storage through dynamic thresholding and shared weight matrices. Adaptive parameter sharing reduces independent parameters, efficiently controlling model complexity without compromising accuracy. Consequently, APSL-BPNN-HMM achieves 32% memory reduction, accelerating convergence and enhancing scalability for large-scale tasks. An embedded subplot illustrates accuracy fluctuations across iterations and memory usage, showing APSL-BPNN-HMM and baseline HMM performance behavior. APSL-BPNN-HMM maintains higher accuracy while optimizing memory utilization. Yellow and blue markers indicate peak accuracies: APSL-BPNN-HMM (96%) and baseline HMM (75%), corresponding to their memory usage at that iteration. An enlarged contour plot emphasizes accuracy peaks for both models, with warmer colors indicating higher accuracy. APSL-BPNN-HMM achieves 96% peak accuracy at approximately 15.15 GB, while HMM reaches 75% at around 20.85 GB.

5.2 Metrics

5.2.1 Classification metrics: Recall, precision and F-score

To compute F-measure, recall and precision calculations are essential. Precision defines the ratio of correctly identified words to all recognized words (Equation 25). For example, if ten speech features are identified as positive, precision measures transformation accuracy to correct textual information. Recall quantifies the percentage of specified keywords identified relative to all keywords that should have been identified (Equation 26). If 10 positive samples exist, recall measures classifier effectiveness in identifying correct features. F-score is the harmonic mean of recall and precision (Equation 27). These metrics utilize four classes: True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN),[37] defined as: **i) True Positive (TP)**: Words present in audio are accurately retrieved as text (e.g., "living" in audio → "living" in text). **ii) False Positive (FP)**: Words not in audio are retrieved as correct words (e.g., "Emanuel run the show" → "E manual run the show," where "E Manual" doesn't exist in audio). **iii) False Negative (FN)**: Words in audio are not correctly retrieved (e.g., "geographical" and "transmission" → "geografical" and "transmition"). **iv) True Negative (TN)**: Words absent in audio are not retrieved as text (e.g., "Hope" absent in audio and not transcribed).

$$\text{Precision} = \frac{TP}{TP + FP} \tag{25}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{26}$$

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{27}$$

Accuracy: Accuracy is calculated based on automatically trained words. For example, "joy" was not in the corpus but phonemes were automatically trained using HMM and retrieved. Measures considered: of total words in audio (A), how many are exactly present (A^+), how many were automatically trained (A^*), and how many were not identified (A')? (Equation 28)

$$\text{Accuracy} = \frac{A^+ + A^*}{A} \times 100 \tag{28}$$

5.3 Error metrics

5.3.1 BLEU score calculation

The Bilingual Evaluation Understudy (BLEU) score evaluates machine-translated text quality against reference translations as shown in Equation 29:

$$\text{BLEU} = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{29}$$

where BP = Brevity Penalty (penalizes short translations), w_n = Weight for n-gram precision, p_n = Precision for n-grams. BLEU scores range from 0 to 100, with higher values indicating better quality.

5.3.2 WER score calculation

Word Error Rate (WER) evaluates Automatic Speech Recognition (ASR) systems and is given by the Equation 30:

$$WER = \frac{S + D + I}{N} \tag{30}$$

where S = Number of substitutions, D = Number of deletions, I = Number of insertions, N = Number of words in the reference.

6. Results

Figure 4 illustrates the ASPL-BPNN-HMM model's accuracy progression over 160 training epochs. The cyan dashed line represents smoothed training accuracy, while the dark blue dotted line represents smoothed testing accuracy. Both curves show rapid accuracy increases during initial epochs, stabilizing after approximately 40 epochs. Training accuracy approaches 100%, while testing accuracy stabilizes slightly below 95%, indicating strong performance with minimal overfitting.

Figure 5 depicts the distribution of recall, precision, and F-score metrics across three distinct categories. These metrics are calculated across the overall dataset of 2000 files, with values varying within three defined percentage ranges: 1) 90-98%, 2) 87-95%, and 3) 87-94%. The Violin plots in Figure 5 showcase the probability density of metric values within the specified percentage ranges. The width of each 'violin' represents the density of values at different levels, with broader sections indicating higher density. The heatmap in Figure 5 illustrates the correlation among these metrics. It provides a visual representation of how these metrics are interrelated, with warmer colors indicating stronger positive correlations and cooler colors indicating negative correlations. This exhibit offers insights into the general trends and relationships within the specified percentage intervals, enhancing our understanding of the dataset's characteristics. The average recall is 95.7%, the precision is 92.95%, and the F-score is 94.53%.

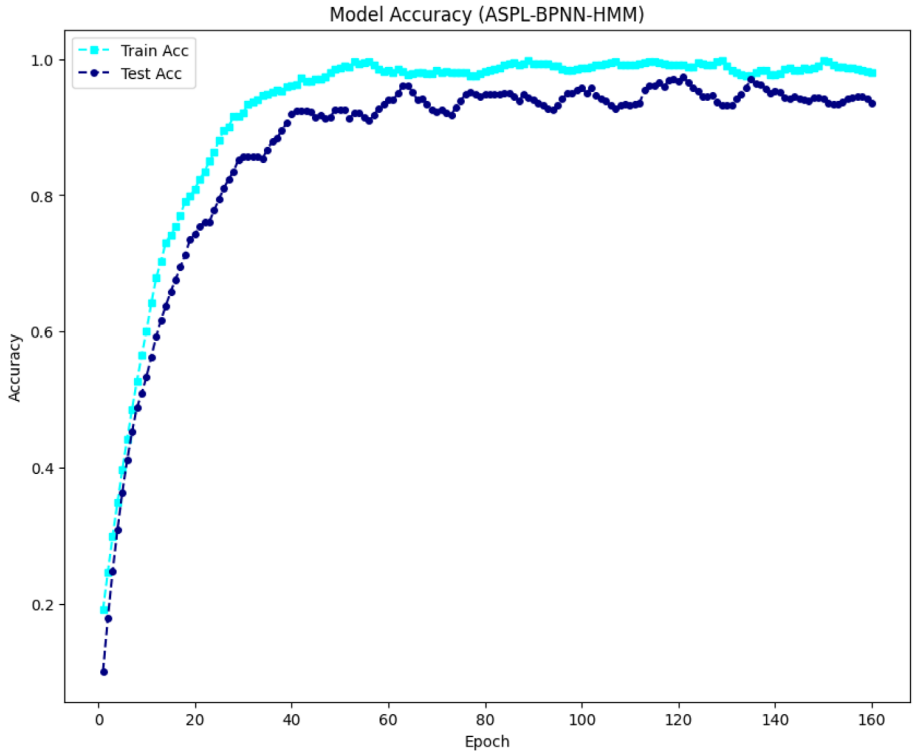


Figure 4. A line graph showing the accuracy trends of the ASPL-BPNN-HMM model across 160 training epochs. The cyan dashed line indicates smoothed training accuracy, which rises quickly and nears 100%. The dark blue dotted line shows smoothed testing accuracy, increasing rapidly in early epochs and leveling off just below 95%.

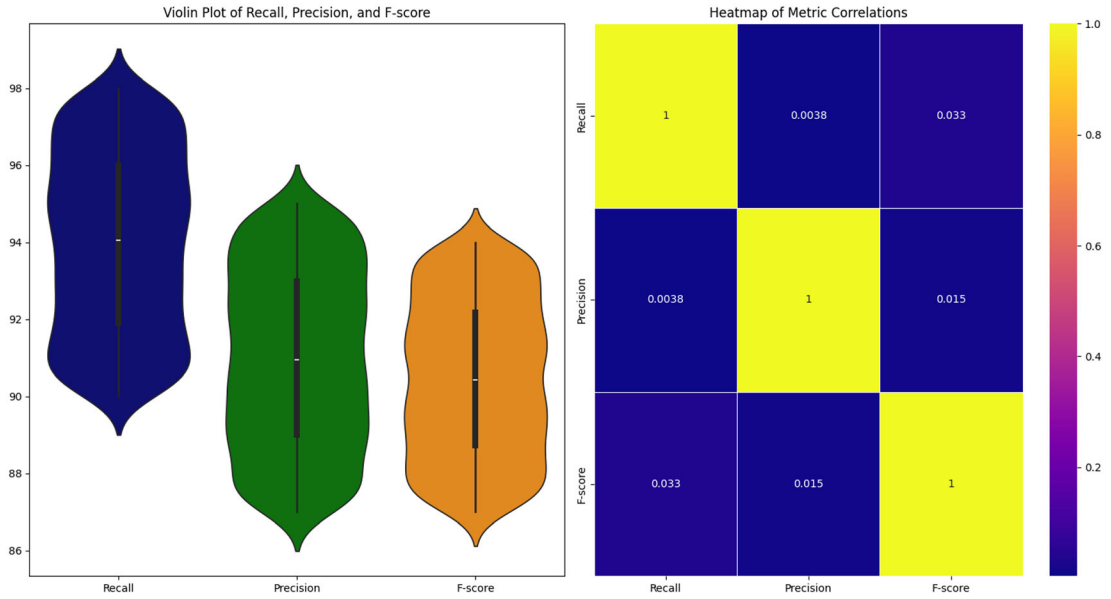


Figure 5. Distribution of performance metrics and correlation analysis of ASPL-BPNN-HMM Model. The left subplot presents a violin plot illustrating the distribution of Recall, Precision, and F-score across defined percentage ranges, while the right subplot displays a confusion matrix highlighting the correlation among these metrics.

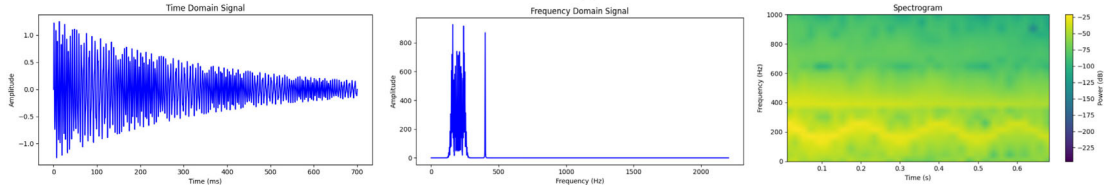


Figure 6. Noisy audio input used for testing the APSL-BPNN-HMM model, HMM model, and Human performance.

To evaluate the performance of our proposed APSL-BPNN-HMM model against the Human and HMM models, we included an audio file containing noise and disturbances. This audio sample served as the input for all models, allowing us to assess their robustness in handling real-world noisy conditions. The noisy input audio file, depicted in Figure 6, reflects typical background noise scenarios such as murmurs in a cafeteria, constant hums from air conditioning units, and random disturbances like keyboard taps or coughs.

Table 3 illustrates the performance of the APSL-BPNN-HMM model in terms of Word Error Rate (WER) and BLEU score for audio recordings collected across diverse geographical regions, as discussed in Section 5.1. The upper portion of the table summarizes the ASR WER, where lower values represent improved recognition accuracy. The lower portion presents the BLEU score for audio translation, where higher scores indicate better translation fidelity. Across all regional categories, the APSL-BPNN-HMM model consistently outperforms the baseline HMM model, narrowing the gap with human transcription and translation performance, which serves as a reference benchmark.

The results of this evaluation, shown in Figure 7, compare APSL-BPNN-HMM, HMM, and Human performance across five filtering conditions and Word Error Rate (WER). Each subplot shows performance trends as the filtering parameter varies (Hz), highlighting the impact of noise-reduction techniques on accuracy and WER. Control and Core Filtering combines fundamental noise reduction with adaptive mechanisms to suppress noise while preserving essential features, e.g., steady background noise in a cafeteria. APSL-BPNN-HMM maintains high accuracy across parameters, whereas HMM declines sharply after parameter 3, and Human performance remains low. Core Spectral Notch Filtering targets specific frequency bands, e.g., removing 60 Hz AC hum in a conference call. APSL-BPNN-HMM performs best at higher parameter values; HMM deteriorates with aggressive filtering, and Humans show declining accuracy. Spectral Notch Filtering applies frequency-specific filtering without adaptivity, e.g., reducing low-frequency hum in a studio podcast.

Table 3. Performance of ASR and translation models across geographical regions.

Geographical region/Metric	Human	HMM	APSL-BPNN-HMM
ASR Word Error Rate (WER) – Lower is Better			
Western European	6	3	2
Eastern European	14	6	3
Central Asia/Middle East/North Africa	21	11	5
Sub-Saharan Africa	33	17	7
South Asia	35	22	8
South East Asia	9	5	3
CJK (CER)	5	5	3
BLEU Score – Higher is Better			
Overall Translation Quality (BLEU Score)	29	40	48

Table 4. Comparison of HMM and APSL–BPNN–HMM performance across five corpora.

Corpus	Model	Accuracy	Precision	Recall	F1-score
Corpus 1	HMM	0.3800	0.8250	0.2200	0.3474
	APSL–BPNN–HMM	0.7250	0.7654	0.9133	0.8328
Corpus 2	HMM	0.4150	0.7143	0.3667	0.4846
	APSL–BPNN–HMM	0.7150	0.7514	0.9267	0.8299
Corpus 3	HMM	0.4650	0.7590	0.4200	0.5408
	APSL–BPNN–HMM	0.7200	0.7640	0.9067	0.8293
Corpus 4	HMM	0.4450	0.7600	0.3800	0.5067
	APSL–BPNN–HMM	0.7500	0.7500	1.0000	0.8571
Corpus 5	HMM	0.4500	0.7381	0.4133	0.5299
	APSL–BPNN–HMM	0.7000	0.7586	0.8800	0.8148

APSL-BPNN-HMM balances noise reduction and signal preservation, HMM struggles at high parameters, and Human performance stays lowest. Core Temporal Notch Filtering integrates core filtering with temporal suppression to handle transient noise, e.g., coughs or keyboard taps. APSL-BPNN-HMM maintains high accuracy; HMM declines with aggressive filtering, and Humans steadily decline. Temporal Notch Filtering targets time-based noise, e.g., chair movements or pen drops in a conference room. APSL-BPNN-HMM shows superior adaptability, HMM deteriorates at high parameters, and Human accuracy remains low. Word Error Rate (WER) measures incorrect words in speech recognition, with lower values indicating better performance. APSL-BPNN-HMM achieves the lowest WER, especially with larger test samples, followed by HMM and then Humans. Overall, APSL-BPNN-HMM consistently outperforms HMM and Humans across all filtering methods, demonstrating robust noise suppression, improved speech recognition, and resilience under aggressive filtering. Its low WER confirms stability and scalability in large-scale evaluations.

Table 4 presents a detailed comparison of the performance of two models — the conventional Hidden Markov Model (HMM) and the proposed APSL-BPNN-HMM — across five representative speech corpora: *1) British National Corpus (BNC),*[27] *2) American National Corpus (ANC),*[28] *3) Corpus of Contemporary American English (COCA),*[29] *4) Buckeye Speech Corpus,*[30] *and 5) Emu Speech Database.*[31] The table reports four key classification metrics for each corpus: Accuracy, Precision, Recall, and F1-Score. Across all corpora, the APSL-BPNN-HMM consistently outperforms the baseline HMM, with notable improvements in recall and F1-score, highlighting its robustness in handling imbalanced and spontaneous speech data. While Buckeye and Emu corpora were partially included in training, rigorous safeguards were implemented to avoid data leakage. Specifically, speaker-level partitioning ensured that no individual's data appeared in both training and testing sets. In addition, temporal segmentation preserved distinct time windows for each data split. Cross-validation techniques were employed to assess generalization, ensuring reliable evaluation of the model on unseen speech samples.

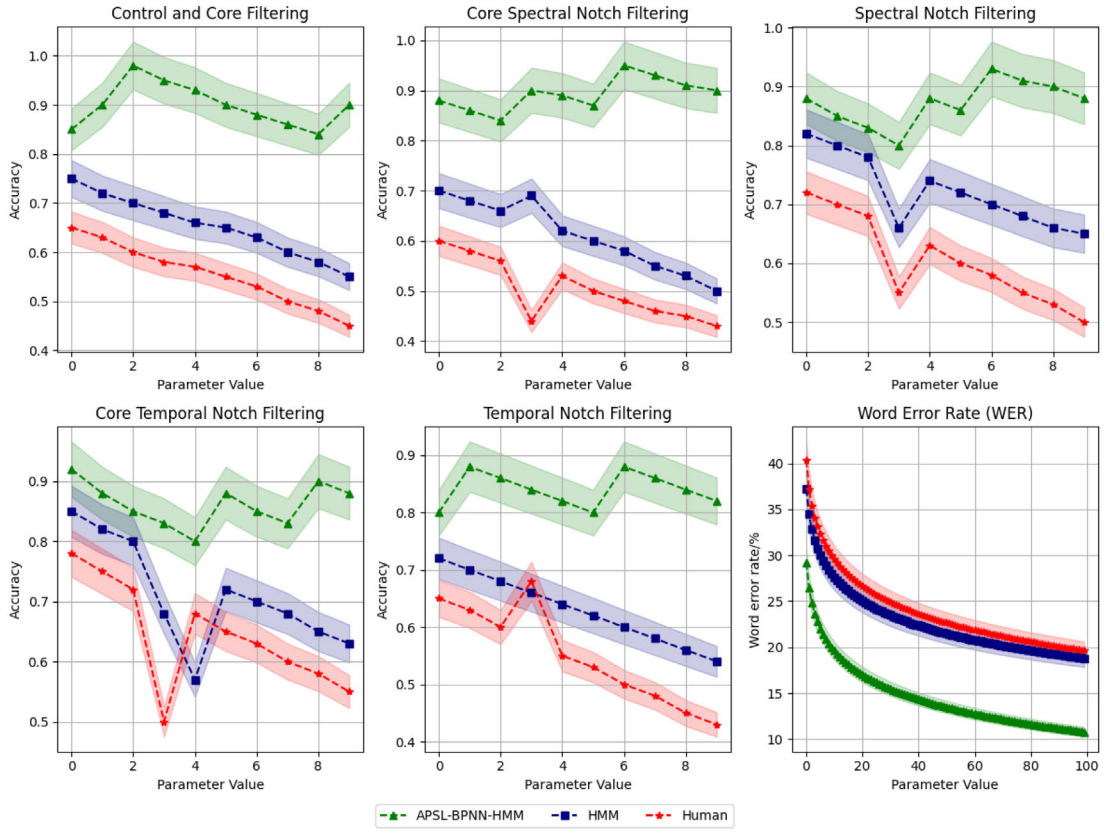


Figure 7. Performance comparison of APSL-BPNN-HMM, HMM, and Human across various noise reduction techniques and Word Error Rate (WER). The subplots represent: (1) Control and Core Filtering, (2) Core Spectral Notch Filtering, (3) Spectral Notch Filtering, (4) Core Temporal Notch Filtering, (5) Temporal Notch Filtering, and (6) Word Error Rate (WER). The shaded regions indicate a ±5% uncertainty range around the plotted values, representing potential variability in the measurements.

7. Discussion

The proposed system translates acoustic features into language models, showing promise for effective speech recognition. However, certain phonemes, such as in "geographical" and "transmission," were misidentified due to errors in mapping acoustic features, leading to syllable and spelling mistakes. Performance is influenced by diverse speaking styles and speaker-listener dynamics—including formal, informal, fearful, threatening, and intimate modes—which interact with psychological aspects of speech. The model adapts to unseen data, while corpus size affects memory requirements: larger dictionaries demand more resources, smaller ones are more efficient. Pronunciation variations in common names and dialect differences, such as US vs. UK standards, add complexity. Latency ranges from 3–5 seconds for typical inputs and 8–10 seconds for complex files, with a word error rate of 10%, indicating efficient recognition of out-of-corpus words. The ASPL-BPNN-HMM approach enhances phoneme identification and sequence mapping but faces challenges. Its complexity requires substantial computational power and hyperparameter tuning, including feature weights and network depth. Noise interference can degrade speech clarity, especially when background sounds mimic key phonemes. Balancing improved recognition with real-time latency remains critical. Despite these issues, ASPL shows strong potential when combined with noise reduction and optimized hyperparameters. Future enhancements include developing models using linguistic features with LSTM for faster text conversion, testing resilience to white Gaussian noise, expanding the database with diverse speaking styles, tuning HMM parameters (states, window size, cepstral coefficients), and evaluating performance on multiple languages to broaden applicability.

8. Conclusion

The realm of phonetics delves beyond mere phonemes, symbols, and sound utterances. It lays the crucial groundwork for mastering phonetic skills by intertwining sounds and characters. This proficiency serves as a springboard for exploring a diverse array of linguistic theories and applications, including speech recognition, speech synthesis, and discourse language transmission. In this context, the current research paper has been a testament to the empirical realization of a

speech-to-text model, representing a significant stride in the field of speech recognition. The proposed methodologies have been meticulously executed on the simulation platform, Praat. The implementation unfolds across various pivotal stages, spanning from speech acquisition to feature extraction. Of note is the experimental elucidation of speech-pause detection, accomplished through an energy-based approach, as well as the feature extraction process employing framing and a window-based method embedded within the Hidden Markov Model (HMM) framework. The outcomes of these experiments have been rigorously scrutinized through established performance metrics, affirming that the acoustic modeling employed in the speech-to-text process attains an impressive level of efficacy through the utilization of HMM. This research paves the way for advanced developments in the realm of speech recognition and showcases the potential of harnessing acoustic modeling techniques for robust and efficient speech-to-text transformation.

Data availability statement

The datasets used in this research are publicly available and can be accessed from the following sources: the British National Corpus (BNC),[27] the American National Corpus (ANC),[28] the Corpus of Contemporary American English (COCA),[29] the Buckeye Speech Corpus,[30] and the Emu Speech Database.[31] The trained model files and derived artefacts generated during the current study are not publicly hosted due to storage and maintenance constraints. However, these materials can be made available for academic and non-commercial research purposes upon reasonable request. Any additional in-house developed datasets and the model developed in this study are available from the corresponding author upon reasonable request. Interested readers and reviewers may apply for access by contacting the corresponding author at r.siddalingappa@yorksj.ac.uk. Access will be granted subject to intended use being consistent with academic research and applicable data usage agreements.

References

1. Hanumanthappa M, Rashmi S, Jyothi NM: **Impact of phonetics in natural language processing: A literature survey.** *IJJISET–International Journal of Innovative Science, Engineering & Technology.* 2014; **1**(3).
2. Patel I, Srinivasa Rao Y: **Speech recognition using hidden markov model with mfcc-subband technique.** *2010 international conference on recent trends in information, telecommunication and computing.* IEEE; 2010; pages 168–172.
3. Le VB, Besacier L, Schultz T: **Acoustic-phonetic unit similarities for context dependent acoustic model portability.** *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings.* IEEE; 2006; volume **1**: pages I–I.
4. Shivakumar KM, Jain VV, Krishna Priya P: **A study on impact of language model in improving the accuracy of speech to text conversion system.** *2017 International Conference on Communication and Signal Processing (ICCSP).* IEEE; 2017; pages 1148–1151.
5. Gunawan A, *et al.*: **English digits speech recognition system based on hidden markov models.** *Proceedings of International Conference Computer.* 2010.
6. Katre SM, *et al.*: *Aṣṭādhyāyī of Pāṇini.* Motilal Banarsidass Publ; 1989.
7. Vijayalakshmi P, Ramani B, Actlin Jeeva MP, *et al.*: **A multilingual to polyglot speech synthesizer for indian languages using a voice-converted polyglot speech corpus.** *Circuits, Systems, and Signal Processing.* 2018; **37**: 2142–2163.
Publisher Full Text
8. Ling Z-H, Zhou X, King S: **The blizzard challenge 2021.** *Proc. Blizzard Challenge Workshop.* 2021.
9. Mutalib NSA, Noah SA: **Phonetic coding methods for malay names retrieval.** *2011 International Conference on Semantic Technology and Information Retrieval.* IEEE; 2011; pages 125–129.
10. Ogbureke KU, Carson-Berndsen J: **Framework for cross-language automatic phonetic segmentation.** *2010 IEEE International Conference on Acoustics, Speech and Signal Processing.* IEEE; 2010; pages 5266–5269.
11. Juneja A, Espy-Wilson C: **Acoustic-phonetic approach to speech recognition based on event detection and linear discriminant analysis.** *J. Acoust. Soc. Am.* 2001; **109**(5_Supplement): 2493–2493.
12. Khanagha V, Daoudi K, Pont O, *et al.*: **Improving text-independent phonetic segmentation based on the microcanonical multiscale formalism.** *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE; 2011; pages 4484–4487.
13. Gales M, Young S, *et al.*: **The application of hidden markov models in speech recognition.** *Foundations and Trends® in Signal Processing.* 2008; **1**(3): 195–304.
Publisher Full Text
14. Mullah HU, Pyrtuh F, Joyprakash Singh L: **Development of an hmm-based speech synthesis system for indian english language.** *2015 international symposium on advanced computing and communication (ISACC).* IEEE; 2015; pages 124–127.
15. Kumar R, Videla LS, SivaKumar S, *et al.*: **Murmured speech recognition using hidden markov model.** *2020 7th International Conference on Smart Structures and Systems (ICSSS).* IEEE; 2020; pages 1–5.
16. Kannamal E, *et al.*: **Investigation of speech recognition system and its performance.** *2020 International Conference on Computer Communication and Informatics (ICCCI).* IEEE; 2020; pages 1–4.
17. Siddalingappa R, Lakshmi BA, *et al.*: **Fedge: Federated learning at the edge on space platforms using deep neural network architectures.** *Int. J. Inf. Technol.* 2025; 1–12.
Publisher Full Text
18. Shuo Zhang L, Liu, and Dingyu Xue.: **Nyquist-based stability analysis of non-commensurate fractional-order delay systems.** *Appl. Math. Comput.* 2020; **377**: 125111.
Publisher Full Text
19. Rashmi S, Hanumanthappa M, Gopala B: **Training based noise removal technique for a speech-to-text representation model.** *Journal of Physics: Conference Series.* IOP Publishing; 2018; volume **1142**: page 012019.
20. Martynova EV, Eremeeva GR, Valieva GF: **The graphical method of pauses detection in english speech signals.** *Utopía y Praxis Latinoamericana.* 2019; **24**(6): 26–31.
21. Boersma P, Van Heuven V: **Speak and unspeak with praat.** *Glot International.* 2001; **5**(9/10): 341–347.
22. Logan B, *et al.*: **Mel frequency cepstral coefficients for music modeling.** *Ismir.* Plymouth, MA: 2000; volume **270**: page 11.
23. Manchanda S, Gupta D: **Hybrid approach of feature extraction and vector quantization in speech recognition.** *Proceedings of the Second International Conference on Computational Intelligence and Informatics: ICCII 2017.* Springer; 2018; pages 639–645.
24. Agarwalla S, Sarma KK: **Machine learning based sample extraction for automatic speech recognition using dialectal assamese speech.** *Neural Netw.* 2016; **78**: 97–111.
PubMed Abstract | Publisher Full Text
25. Rashmi S, Hanumanthappa M, Reddy MV: **Hidden markov model for speech recognition system—a pilot study and a naive approach for speech-to-text model.** *Speech and Language Processing for Human-Machine Communications: Proceedings of CSI 2015.* Springer; 2018; pages 77–90.
26. Wong PHW, Au OC, Wong JWC, *et al.*: **Reducing computational complexity of dynamic time warping-based isolated word recognition with time scale modification.** *ICSP'98. 1998 Fourth*

International Conference on Signal Processing (Cat. No. 98TH8344). IEEE; 1998; pages 722–725.

27. Aston G, Burnard L: *The BNC handbook: exploring the British National Corpus with SARA.* Edinburgh University Press; 2020.

28. Ide N, Macleod C: **The american national corpus: A standardized resource of american english.** *Proceedings of corpus linguistics.* Lancaster University Centre for Computer Corpus Research on Language; 2001; volume **3**: pages 1–7.

29. Davies M: **The 385+ million word corpus of contemporary american english (1990–2008+): Design, architecture, and linguistic insights.** *International journal of corpus linguistics.* 2009; **14**(2): 159–190.

30. Pitt MA, Johnson K, Hume E, *et al.*: **The buckeye corpus of conversational speech: Labeling conventions and a test of transcriber reliability.** *Speech Comm.* 2005; **45**(1): 89–95.

31. Cassidy S, Harrington J: **Multi-level annotation in the emu speech database management system.** *Speech Comm.* 2001; **33**(1-2): 61–77.
Publisher Full Text

32. Hecht-Nielsen R: **Theory of the backpropagation neural network.** *Neural networks for perception.* Elsevier; 1992; pages 65–93.

33. Forney GD: **The viterbi algorithm.** *Proc. IEEE.* 2005; **61**(3): 268–278.
Publisher Full Text

34. Rechkemmer A, Yin M: **When confidence meets accuracy: Exploring the effects of multiple performance indicators on trust in machine learning models.** *Proceedings of the 2022 chi conference on human factors in computing systems.* 2022; pages 1–14.

35. Snoek J, Larochelle H, Adams RP: **Practical bayesian optimization of machine learning algorithms.** *Adv. Neural Inf. Proces. Syst.* 2012; **25**.

36. Conneau A, Ma M, Simran Khanuja Y, *et al.*: **Fleurs: Few-shot learning evaluation of universal representations of speech.** *2022 IEEE Spoken Language Technology Workshop (SLT).* IEEE; 2023; pages 798–805.

37. Siddalingappa R, Kanagaraj S: **Anomaly detection on medical images using autoencoder and convolutional neural network.** *Int. J. Adv. Comput. Sci. Appl.* 2021; **12**.
Publisher Full Text

Open Peer Review

Current Peer Review Status: ✗ ✗ ✓

Version 1

Reviewer Report 06 May 2026

https://doi.org/10.5256/f1000research.195635.r479270

✓ **Ramanda Rizky**
Universitas Lancang Kuning, Pekanbaru, Riau, Indonesia

This paper presents a hybrid Automatic Speech Recognition (ASR) framework that combines a Backpropagation Neural Network (BPNN) with a Hidden Markov Model (HMM), enhanced by an Adaptive Phoneme State Learning (APSL) mechanism. In terms of structure, this study is organized according to conventional scientific standards, proceeding systematically from preprocessing and feature extraction to modeling and evaluation. This structure supports readability and demonstrates a coherent research workflow. However, a critical review reveals several fundamental issues that limit the scientific robustness of this study, particularly regarding the relevance of benchmarks, theoretical foundation, reproducibility, and the validity of its conclusions.

The primary concern lies in the benchmarking strategy. Although this study demonstrates that the proposed APSL-BPNN-HMM model outperforms conventional HMM baselines on metrics such as precision, recall, F1 score, and Word Error Rate, this comparison is insufficient in the context of contemporary ASR research. The field has undergone a paradigm shift toward deep learning and end-to-end architectures, including Transformer-based models, Connectionist Temporal Classification (CTC), and Recurrent Neural Network Transducers (RNN-T). By limiting the evaluation to traditional HMM baselines and, unusually, human transcriptions under noisy conditions, this study does not provide a meaningful frame of reference for assessing its contributions. Consequently, the claimed performance improvements lack external validity. To be scientifically valid, this study must include comparisons with modern ASR systems and position its contributions relative to current state-of-the-art approaches.

Similarly important are issues of theoretical rigor. The APSL mechanism is introduced as the core innovation of this study, yet its formulation is largely procedural rather than analytical. Although the paper provides equations and step-by-step descriptions, the APSL mechanism is not clearly situated within a well-defined probabilistic or machine learning framework. The adaptive adjustment of phoneme transition probabilities using neural confidence scores appears heuristic, with limited justification grounded in established theory. This weakens both the interpretability and generalizability of the approach. A scientifically valid contribution requires not only functional implementation but also a clear theoretical foundation explaining why and under what conditions the method should work. Reinforcing this aspect involves formal derivations, clearer assumptions,

and explicit connections to existing probabilistic adaptation or hybrid modeling techniques. Moreover, the reproducibility represents another critical limitation. Although this study describes a general workflow and reports some hyperparameters, it does not provide sufficient detail to allow for full replication. Key aspects of the experimental setup remain unclear, including the exact composition of the training, validation, and test splits; the proportions and configuration of the augmented data; and the specific preprocessing applied to each dataset. Furthermore, the absence of publicly available code, trained models, or configuration files further limits reproducibility. While the use of publicly accessible corpora is a positive step, it is insufficient on its own. In contemporary empirical research, reproducibility is closely tied to transparency, and this generally requires open access to implementation resources. Addressing this issue would significantly enhance the credibility and impact of this work.

Dealing with data availability in this study is also incomplete. Although this study utilizes established corpora such as BNC, ANC, and COCA, the integration of these datasets with expanded data and internally generated data is not fully documented. Without clear documentation on how these datasets were combined, preprocessed, and balanced, it will be difficult for other researchers to replicate the experimental conditions or verify the reported results. Providing a detailed data protocol, including preprocessing scripts and augmentation procedures, would help bridge this gap and align this research with best practices in open and reproducible science.

Another important limitation concerns the interpretation of results and the strength of conclusions. The findings consistently show that the proposed model outperforms the baseline HMM, which supports the internal validity of this study. However, the conclusions drawn go beyond what the available evidence can support. Claims implying performance approaching human levels or broader application to real-world ASR scenarios are not adequately supported, particularly given the lack of comparison with modern systems and the absence of statistical validation. Most of the analysis is descriptive, relying on average performance metrics without reporting variance, confidence intervals, or significance tests. This makes it difficult to determine whether the observed improvements are robust or merely coincidental. For conclusions to be scientifically justified, they must be more aligned with the scope and limitations of the experimental design.

Broadly speaking, these issues highlight four core areas that must be addressed for this study to achieve scientific validity. First, the benchmarking framework must be expanded to include contemporary ASR models, ensuring that performance claims are evaluated against relevant standards. Second, the APSL mechanism requires a stronger theoretical foundation, moving beyond heuristic descriptions toward formal justification. Third, the study must improve reproducibility by providing detailed methodological documentation and, ideally, open access to code and data processing workflows. Fourth, conclusions should be moderated and supported by more rigorous analysis, including appropriate validation techniques. Addressing these areas will not only strengthen the internal coherence of the study but also enhance its relevance, credibility, and contribution to the evolving field of speech recognition research.

Is the work clearly and accurately presented and does it cite the current literature?

Partly

Is the study design appropriate and is the work technically sound?

Partly

Are sufficient details of methods and analysis provided to allow replication by others?
Yes

If applicable, is the statistical analysis and its interpretation appropriate?
I cannot comment. A qualified statistician is required.

Are all the source data underlying the results available to ensure full reproducibility?
Partly

Are the conclusions drawn adequately supported by the results?
Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: English as a Foreign Language, English Language Teaching, Applied Linguistics.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Reviewer Report 06 May 2026

https://doi.org/10.5256/f1000research.195635.r474867

Fayzulla Nazarov
Artificial Intelligence, Samarkand State University named after Sharof Rashidov (Ringgold ID: 187914), Samarkand, Samarkand Province, Uzbekistan

Ilyos Khujayorov
Artificial Intelligence, Tashkent University of Information Technologies named after Muhammad al-Khwarizm (Ringgold ID: 187932), Tashkent, Tashkent Province, Uzbekistan

The article proposes the APSL algorithm combining traditional BPNN and HMM models.
The following deficiencies and suggestions were identified during the review:

1. The introduction part started with basic information related to explaining general definitions and concepts. The introduction part of scientific articles must give information like Research Context motivation, research Aims, Contributions, and others. In short, the modern problem (Problem Statement) and relevance are not revealed.
2. In the literature review part, mainly sources from 15-20 years ago are presented as main researches, but this does not reflect the current state of this field. It is appropriate to emphasize articles published within the last 3-5 years (modern State-of-the-Art models are left out of attention). Authors should completely revise the literature review, it is recommended to add publications between the years 2022–2026, especially to scientifically

justify the difference between End-to-End models (Whisper, Conformer) and HMM, and to clearly show the place of the article in the era of these technologies.
3. There is no information about the hardware used in Training process.
4. How many states were used for each phoneme (for example, standard 3-state HMM or otherwise) is not clearly stated in the methodology part. This is one of the most important parameters of acoustic modeling.
5. The dynamic windowing (Eq. 18) process is not clarified. A proposal is given to increase the window size by +5 ms when the confidence coefficient is low. Theoretical or experimental bases are not presented about why it is exactly 5 ms and how this affects the time delay.
6. It is stated that the APSL mechanism reduced memory from 24 GB to 15.12 GB. However, an analysis proving that such a reduction did not negatively affect accuracy should be given in more detail in the methodology part.
7. Comparison works of the model results proposed by the authors with modern E2E architectures have not been done. This is considered one of the important issues of checking the reliability of the model. Authors must justify why they chose exactly the HMM-BPNN hybrid compared to Whisper or other modern models.
8. The mathematical expression of the APSL algorithm (17) has a heuristic appearance and its theoretical basis is not sufficiently revealed.
9. The conclusion part of the article is written in a very general way. It is appropriate to separately note the most important numerical indicators achieved in the conclusion, the role of the APSL algorithm in saving memory, and also add thoughts regarding real-time requirements.
Due to the serious technical and methodological deficiencies noted above, I recommend rejecting this article for publication.

Is the work clearly and accurately presented and does it cite the current literature?
Yes

Is the study design appropriate and is the work technically sound?
Partly

Are sufficient details of methods and analysis provided to allow replication by others?
Partly

If applicable, is the statistical analysis and its interpretation appropriate?
Partly

Are all the source data underlying the results available to ensure full reproducibility?
No

Are the conclusions drawn adequately supported by the results?
Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Digital signal processing, NLP, speech recognition and synthesis,AI, parallel computing

We confirm that we have read this submission and believe that we have an appropriate level of expertise to state that we do not consider it to be of an acceptable scientific standard, for reasons outlined above.

Reviewer Report 13 March 2026

https://doi.org/10.5256/f1000research.195635.r464331

Hamza Kheddar
University of Medea,, Medea, Algeria

The topic is interesting; however, the paper needs significant improvement:

- The proposed APSL-BPNN-HMM architecture relies on classical models (BPNN and HMM) and does not sufficiently justify its advantages compared to modern deep learning ASR frameworks such as Transformer-based or end-to-end models (e.g., CTC, RNN-T). This limits the perceived novelty and relevance of the work in the current ASR research landscape.

read and compare with the following for example:

Deep Transfer Learning for Automatic Speech Recognition: Towards Better Generalization

Machine learning approaches for automated detection and classification of dysarthria severity

Noise-robust speech recognition: A comparative analysis of LSTM and CNN approaches

A robust framework for noisy speech recognition using Frequency-Guided-Swin Transformer

- The description of the Adaptive Phoneme State Learning (APSL) algorithm lacks rigorous mathematical formalization and theoretical justification. Several steps appear heuristic, and the derivation of the adaptive transition probabilities is not clearly justified or compared with existing probabilistic adaptation methods.

- The experimental evaluation compares the proposed method mainly against a traditional HMM baseline and human transcription. However, the study does not include comparisons with contemporary ASR systems (e.g., deep neural acoustic models or end-to-end models), making it difficult to assess the true competitiveness of the proposed approach.

- Although multiple speech corpora are used, the experimental protocol and data splitting strategy are not described in sufficient detail. The use of augmented data and partially overlapping corpora raises concerns about possible bias or insufficient independence between training and testing

sets.

- The paper claims scalability and real-time applicability, yet the architecture involves multiple processing stages (MFCC extraction, APSL segmentation, BPNN classification, and HMM decoding). The computational cost and latency are only briefly discussed (3–10 seconds for recognition), which may limit real-time deployment.

- The authors acknowledge that pronunciation variations, dialect differences, and background noise can degrade performance, leading to phoneme misidentification in words such as “geographical” or “transmission.” This suggests the model may struggle with complex linguistic variability and real-world acoustic conditions.

- Most all references are old-dated

Is the work clearly and accurately presented and does it cite the current literature?
Yes

Is the study design appropriate and is the work technically sound?
Partly

Are sufficient details of methods and analysis provided to allow replication by others?
Partly

If applicable, is the statistical analysis and its interpretation appropriate?
Partly

Are all the source data underlying the results available to ensure full reproducibility?
No

Are the conclusions drawn adequately supported by the results?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: The methodological contribution appears incremental rather than fundamentally novel

I confirm that I have read this submission and believe that I have an appropriate level of expertise to state that I do not consider it to be of an acceptable scientific standard, for reasons outlined above.