GRADIVA

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

Performance Analysis of OFSCNN Model based Dialect Recognition System for Bagri Language

Er. Poonam Kukana [1], Dr. Pooja Sharma [2], Dr. Puneet Sapra [3] & Dr. Neeru Bhardwaj [4]

1,2,3. Department of Computer Science and Engineering, University School of Engineering & Technology,
Rayat-Bahra University, Mohali, Punjab, India.
4. Department of Computer Science and Engineering, Chandigarh University, Mohali, Punjab, India.
Email: [1] poonamkukana@gmail.com, [2]poojashrm27@gmail.com,
[3]puneetsapra91@gmail.com, [4]neeru.apcse@gmail.com

Abstract

In this paper, we propose an Optimized Featured Swarm Convolutional Neural Network (OFSCNN) Model for dialect identification in the Bagri Rajasthani Language. The OFSCNN model combines the power of convolutional neural networks with swarm intelligence to create a highly accurate and efficient system. We trained the proposed system on a large dataset of Bagri Rajasthani Language speech samples and evaluated its performance using various performance metrics. Furthermore, we compared the effectiveness of the proposed OFSCNN model's performance against other dialect recognition methods already in use, such as Multi Support Vector Machine (Multi SVM) and Genetic Algorithm Neural Network (GA-NN). Our experimental results demonstrate that the proposed OFSCNN-based dialect identification system achieved a high level of accuracy, specifically 96.95%. In comparison, the Multi SVM model achieved an accuracy of 93.45%, while the GA-NN model achieved 80.63%. Our OFSCNN model outperformed both the Multi SVM and GA-NN models, showcasing the effectiveness of our approach. Our findings underscore that the proposed OFSCNN model serves as an effective tool for dialect identification in the Bagri Rajasthani Language. This proposed system holds potential applications in speech recognition, machine translation, and speaker identification. Our work contributes to ongoing research in dialect identification and highlights the viability of swarm intelligence-based approaches in natural language processing.

Index Terms: *Dialect identification, Bagri Rajasthani Language, Optimized Featured Swarm Convolutional Neural Network (OFSCNN), Multi Support Vector Machine, Genetic Algorithm Neural Network.*

I. INTRODUCTION

Distinct dialects serve as bridges between people, facilitating communication across various languages. These languages are powerful tools for sharing ideas between individuals. However, it is in their interactions within different contexts that these languages reveal their unique characteristics. The challenge lies in identifying dialects within a language, given the variations in pronunciation, grammar, and vocabulary. In this study, we propose an Optimized Featured Swarm Convolutional Neural Network (OFSCNN) Model for identifying dialects in the Rajasthani Language, specifically focusing on the Bagri dialect.

The proposed OFSCNN model synergizes the capabilities of convolutional neural networks with swarm intelligence, resulting in a system that is both highly accurate and efficient. This model undergoes training on an extensive dataset comprising speech samples in the Bagri Rajasthani Language. Its performance is evaluated through the utilization of various performance metrics.

In the preprocessing stage, voice signals undergo a series of steps to eliminate background noise and silence, and are then divided into frames. Typically, each frame lasts for 20–30 ms with a 50% overlap. Subsequently, Mel Frequency Cepstral Coefficients (MFCCs) are extracted from each frame using a filter bank composed of Mel-scaled triangle filters. These MFCC feature vectors adeptly capture the spectral characteristics of the speech signals. The MFCC used in proposed architecture is based on global feature matrix (all possible features toward a uploaded sample). The reduced feature matrix is in size of 500x500 as an output of MFCC that further used in the optimization process to build final training matrix.

To optimize the separability among different speech classes, Particle Swarm Optimization (PSO) is employed to refine the MFCC feature vectors. The filter bank settings undergo iterative updates via the PSO algorithm until a set of optimized features is achieved. These enhanced MFCC characteristics are then fed into a Convolutional Neural Network (CNN) as input for the purpose of classification.

The CNN architecture comprises multiple convolutional layers, followed by pooling layers, and ultimately culminating in a final layer for classification.

The effectiveness of the proposed OFSCNN model's performance is compared to other existing dialect recognition methods, such as Multi Support Vector Machine (Multi SVM) and Genetic Algorithm Neural Network (GA-NN).

A dialect is a distinct form of language spoken in a specific geographical location, differing from other variants of the same language in terms of pronunciation, syntax, and vocabulary. It is often characterized by speech patterns that deviate from the official language. The task of distinguishing a speaker's regional dialect within a given language is known as dialect identification.

Developing a reliable dialect recognition system for the Rajasthani language holds several significant reasons:

- **Cultural Preservation**: Rajasthani, a language with numerous regional dialects, is spoken in the Indian state of Rajasthan and its surrounding areas. Recognizing and preserving these dialects contributes to the promotion and safeguarding of the cultural heritage of the region.
- **Linguistic Research**: The study of differences among various Rajasthani dialects offers valuable insights into the language's structure and evolution. Utilizing speech recognition for the automatic identification and classification of dialects assists linguists in their research endeavors.
- **Enhanced Communication and Education**: Accurate identification of spoken Rajasthani dialects can facilitate improved communication and education. A proficient speech recognition system capable of identifying dialects can enhance cross-regional communication and aid language education by providing feedback on pronunciation and accent.

In conclusion, Rajasthani language dialect recognition serves as a valuable tool for advancing cultural preservation, linguistic research, effective communication, and education."

While language identification has garnered significant attention, the related issue of dialect identification has not received nearly as much focus. Research in the field of dialects remains limited due to a scarcity of databases and the time-intensive nature of analytical methods. This research presents an application designed to recognize the Rajasthani dialect, showcasing the potential utility of voice-based dialect identification. Several practical applications are outlined below:

- **Forensic Speech Analysis:** Automated dialect recognition can assist forensic speech scientists in speaker profiling tasks, aiding criminal investigations.
- **Criminal Activity Monitoring:** An intelligent surveillance system can monitor suspicious activities and expedite inquiries involving suspects attempting to conceal their identities.
- **Crime Prevention:** Criminals often commit offenses through voicemails and phone calls.
- **Tourist Assistance:** Non-local accents can be identified at tourist help centers, travel services, or automated global call systems.
- **Personalized Assistance**: Recognizing a dialect enables selecting the appropriate assistant who speaks the user's native language.

Dialect identification is a crucial technique for deciphering spoken language. This advancement contributes to improving speech recognition systems in modern electronic devices and supports e-wellbeing and telemedicine services, particularly beneficial for elderly populations.

II. RELATED WORK

Several research papers have been published on dialect recognition systems for various Indian languages. Here's a comparison of some of them:

Speech plays a particularly significant role in machine communication in India, where many individuals with low literacy skills communicate primarily in their spoken native language. The development of voice applications in India is challenging due to the country's multitude of official languages.

In one study, an innovative approach utilizing genetically tuned multilayer perceptrons was employed to create a speech recognition system for ten spoken Hindi digits. Genetic Algorithm (GA) optimization of Multi-Layer Perceptrons (MLPs) led to a 1-2% improvement over conventional methods [1].

For Arabic dialect identification, a multi-class Support Vector Machine (SVM) method was proposed to discriminate between five common Arabic dialects—Egyptian, Gulf, Levantine, North African, and Modern Standard Arabic (MSA)—with an accuracy of 59.2%. The study also investigated dialect identification in Arabic broadcast speech using phonetic, lexical, and acoustic features [2].

In a different approach, unsupervised deep learning techniques were applied to Language Identification/Dialect Identification, resulting in a +58% enhancement in "C average" classification performance. Notably, these improvements were achieved without the use of additional secondary speech corpora [3].

Supervised techniques driven by data, such as deep neural networks, have emerged as competitive alternatives to traditional unsupervised methods. Deep learning techniques were explored for addressing additive and convolutional voice degradation, providing guidance for those working on environmentally robust speech recognition systems [4].

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

Another research study focused on identifying Himachal Pradeshi dialects using MFCC (Mel Frequency Cepstral Coefficients) and Gaussian Mixture Model (GMM). The study covered seven dialects from various parts of Himachal Pradesh and highlighted the challenges in differentiating closely related dialects. Online and offline tests yielded accuracy levels of approximately 80% and 70%, respectively [5].

The dialect of speakers is a crucial aspect of speech information. This study presents a machine learning approach to distinguish dialects from acoustic sonorant sound outputs. The approach incorporates coarticulatory information from neighboring vowels to enhance the accuracy of dialect categorization. The neural network achieved an 81% accuracy in distinguishing between dialects [6].

In another research effort, Modern Standard Arabic phonemes were categorized in isolated words, exploring Hidden Markov Model (HMM) variants with 8, 16, and 32 Gaussian mixtures for phoneme recognition. The three HMM system variants achieved overall accurate rates of 83.29%, 88.96%, and 92.70% for categorizing Dynamic Positioning and Feedback (DPF) element classes [7]. Genetic Algorithms and SVM were applied to a three-class classification problem on compressed subsets, surpassing the challenge baseline in terms of Unweighted Average Recall (UAR) [8].

The proliferation of digital home assistants with voice-activated interfaces owes credit to significant advancements in machine learning and signal processing [9]. Recent speaker identification efforts have predominantly utilized i-vector approaches, but deep learning methods, particularly convolutional neural networks, are rapidly taking over. A novel speaker identification approach employing deep CNN was proposed, demonstrating the network's ability to handle variable-length segments and learn speech features from different elements of input speech [10].

The development of Automatic Speech Recognition (ASR) has seen substantial progress, with effective distributed learning techniques playing a pivotal role in training ASR models [11]. An advanced Hindi ASR system demonstrated superior performance using a discriminatively trained HMM system with 256 Gaussian mixtures. Incorporating a Recurrent Neural Network Language Model (RNN LM) further improved system performance [12].

The effectiveness of experimental algorithms for dialect identification was evaluated on custom word databases for four languages, resulting in an accuracy of 98.1%, a substantial improvement over the baseline accuracy of 82% [13]. Deep learning models have streamlined feature extraction procedures and classification models compared to older models [14].

In a separate study, spectral and prosodic information was extracted for the Kannada language to develop an Automated Dialect Identification (ADI) system, achieving an 86.25% improvement in dialect recognition [15]. The Kaldi toolkit was employed to develop a Punjabi hybrid Deep Neural Network (DNN) speech recognition system, with DNNs significantly enhancing system performance. MFCC features outperformed Perceptual Linear Prediction (PLP) features, and the triphone model exhibited better word mistake rates than the monophone model [16].

Hybrid features and Artificial Neural Network (ANN) were employed for speech recognition, with RASTA-PLP hybrid features yielding the highest classification accuracy of 89.6%. Novel training functions were proposed, evaluated through multiple modeling tests on a linguistic dataset containing Malayalam, English, Tamil, and Hindi [17]. Spoken language identification, despite variations in length, pace, subject, and moderator, aims to identify the language present in spoken samples. The effectiveness of spoken language models was assessed to aid researchers in developing new identification models [18].

Continuing the use of Literary Tamil (LT) alongside Colloquial Tamil (CT) in computer-aided language learning tools preserves LT while enhancing CT's usability. This research explores five strategies based on Gaussian Mixture Model (GMM) dialect identification, achieving an 87% accuracy rate [19]. A comprehensive literature review was conducted to gather information on the unique features of Indian languages [20].

Numerous research papers have been published on dialect recognition systems for various Indian languages.

III. METHODOLOGY

3.1 The Proposed Work Flow

Experimental Setup: The experiments were conducted using MATLAB, a widely used platform for scientific and numerical computing. MATLAB provides a versatile environment for implementing and testing various machine learning and signal processing algorithms. The experiments were executed on a [specify hardware configuration if relevant], ensuring a consistent and controlled environment.

The proposed flowchart, as shown in Fig. 1, consists of various modules aimed at processing and identifying dialects from speech signals.

1. **Dataset:** The dataset utilized for training and testing the classifiers was manually compiled from various sources, containing audio recordings of the Bagri Rajasthani dialect. The dataset was imported and processed using MATLAB's built-in functions, allowing for efficient data handling and manipulation.
2. **Pre-processing**: In the next stage, pre-processing is carried out to eliminate noise and artifacts from the speech signals. Techniques such as filtering, windowing, and normalization may be applied during this step.
3. **Feature Extraction**: Once the speech signals have undergone pre-processing, they are transformed into a suitable format for input into the Convolutional Neural Network (CNN). This transformation might involve creating spectrograms or Mel-Frequency Cepstral Coefficient (MFCC) representations from the voice signals. The subsequent phase involves designing the architecture of the CNN, including decisions regarding the number of Convolutional Layers (CLs), Pooling Layers (PLs), Fully Connected Layers (FCLs), Activation Functions (AFs), and normalization layers to be utilized.
4. **Training**: The extracted features and their corresponding dialect labels are employed to train the CNN. Training involves optimizing the network's parameters to minimize the difference between predicted and actual outputs.
5. **Testing**: After the training phase is complete, a separate test dataset is utilized to evaluate the performance of the CNN. The same pre-processing steps used during training are applied to the test dataset.
6. **Deployment**: The trained CNN can be deployed as part of a speech recognition system for dialect identification. This deployment may involve integrating the CNN with additional components, such as a front-end speech processing unit or a back-end database.

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

3.2 Dataset

Datasets form a critical cornerstone of every research project, and our study involves a meticulously assembled dataset sourced from various origins. Presently, there is no standardized voice database specifically tailored for Rajasthani dialect research in the realm of speech processing.

In this research endeavor, we not only elucidated a method for recognizing the Rajasthani dialect, but we also compiled the essential speech corpora requisite for dialect recognition.

In this study, a text-free Rajasthani dialect voice dataset was collected from diverse locations across Rajasthan, particularly in rural and interior areas. According to research, linguistic cues intrinsic to speaking activities differ from those in reading dialogue. Different speaking rates, filled intervals, loudness, tone, apprehensions, echoes, and half-said phrases, among other things, are all visible prosodic indicators in expressive language. These characteristics are considered to transmit dialectal variations more effectively. In the process of data collection, a real-time voice recorder was employed to gather a comprehensive dataset. This dataset consisted of speeches in the Rajasthani language with each of the ten users contributing fifty speeches. The participants were students from two distinct colleges, providing a diverse range of linguistic expressions for analysis. The utilization of a voice recorder allowed for the capture of authentic and spontaneous speech patterns, adding richness and depth to the dataset.

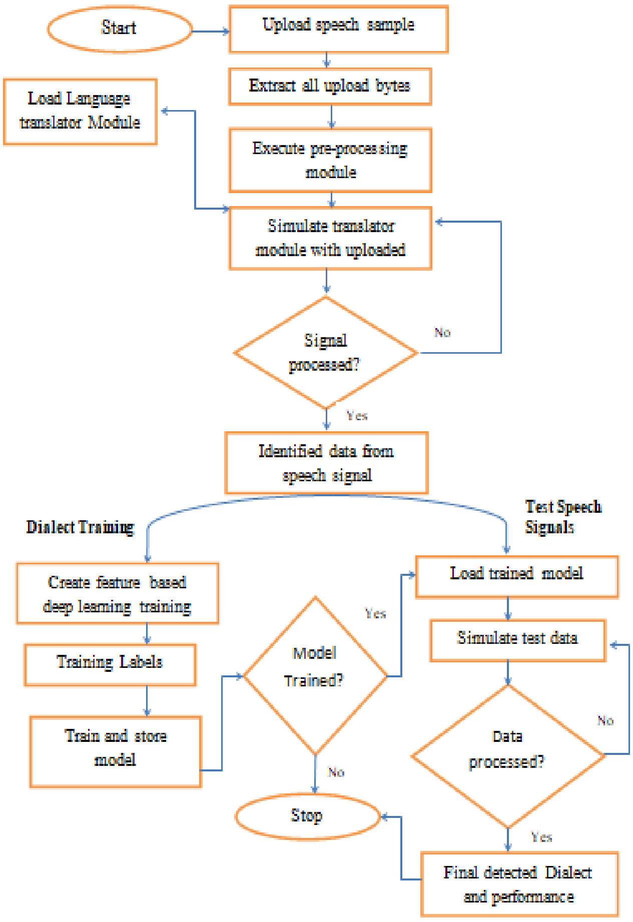

Figure 1: Workflow of Dialect Recognition System

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

The detailed list of sentences can be found in Table 1, encompassing 500 audio recordings featuring 50 distinct sentences uttered by 10 speakers fluent in the Rajasthani dialect. These audio recordings were meticulously captured within a noise-controlled environment using voice recorder software, with the data stored in the.mp3 format.

The dataset encompasses 50 Rajasthani sentences, each comprised of 2 to 4 words. For those interested, the datasets employed in our study can be accessed via the "Speech-Dataset-of-Rajasthani-language" repository, which is hosted by Poonam Kukana [21].

Table 1: Rajasthani Sentences used for the Current Work

Sr. No	Rajasthani Dialect
1	रोटी जीमौ।
2	पाणी पीऔ।
3	इणनै सुणौ।
4	ना जाऔ।
5	अठीनै देकौ।
6	रोटी वणाऔ।
7	बारै जावौ।
8	अठे सुवौ।
9	म्हनैं दिखावौ।
10	मा आऔ
11	अठी आवौ।
12	म्हनै सुणौ।
13	बोळा रौ।
14	धक्कौ देवौ।
15	पाणी उबाळौ ।
16	फुटरौ काम!
17	चूप रौ!
18	गोळी खावौ।
19	बोत बड़ीया!
20	म्हनैं सुणौ।
21	इनै केवौ।
22	वौ म्हारौ है।
23	वौ कठे है?
24	वौ अठे है।
25	इने अठै लावौ।
26	इने वठै मेळौ।
27	आपरौ मुं खोलौ।
28	इणने आछौ राखौ।
29	होलै वात करौ।
30	म्हनै ठा कोनी।
31	खानौ बणगौ हैं।
32	आपरा हाथ धोवौ ।
33	थै थक गिया।
34	वौ कुण है?
35	रोटी कौनी जीमौ ।
36	वौ वठै कौनी।
37	टाबर नै पकड़ौ।
38	थै कठै जावौ?
39	थै कदै आया?
40	कुण रोवै है?
41	म्हनै स़मझ कोनीं आवै।
42	वै म्हारां रिश्तेदार है।
43	थौरै कितरा टाबर है?
44	आज घणी गरमी है।
45	आज सड़क सूखौड़ी है।
46	वौ छोरो रोवै है।
47	आज ठड़ सी है।
48	म्हारै कनै थौड़ाक है।
49	आऔ मौरे साथै जिमौ।
50	आज घणी गरमी है।

3.3 Flow of Current Work

Figure 2 illustrates the descriptive flow of the OFSCNN, Multi-SVM, and GA-NN models for Dialect Identification in the Bagri Rajasthani Language.

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

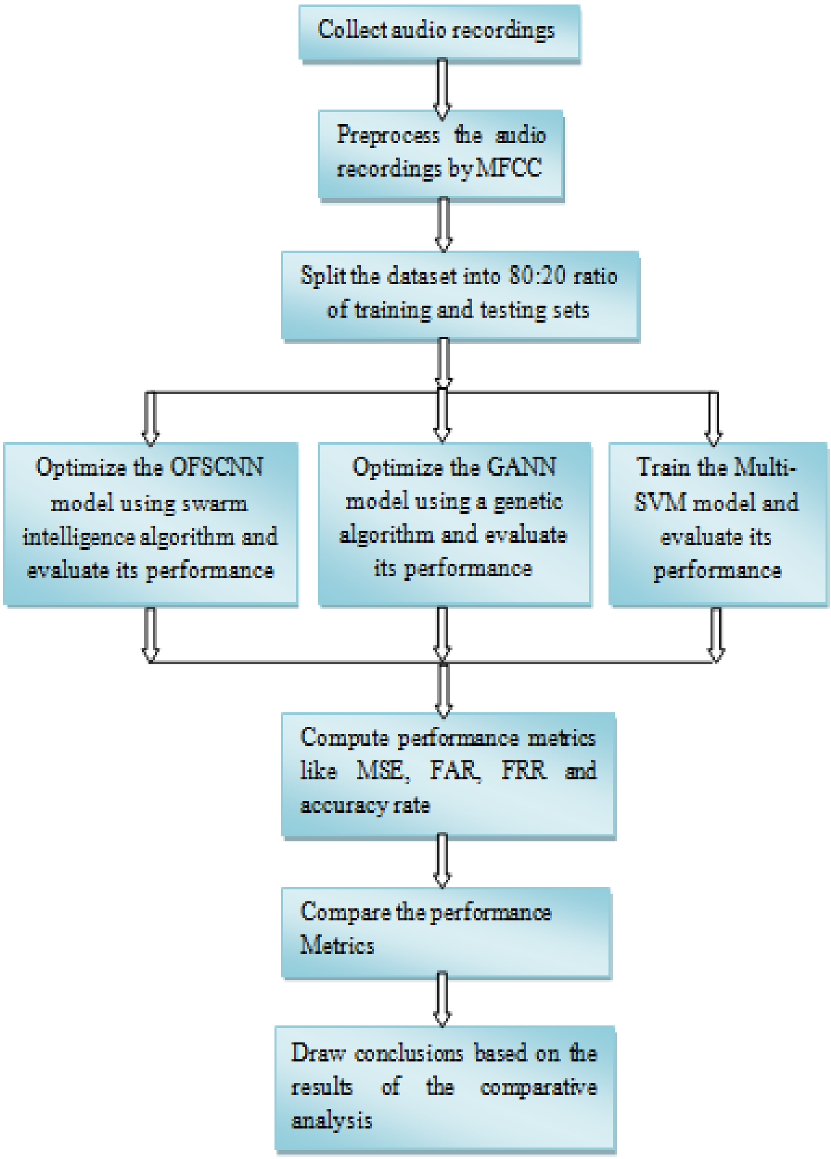
Collect audio recordings
Preprocess the audio recordings by MFCC
Split the dataset into 80:20 ratio of training and testing sets
Optimize the OFSCNN model using swarm intelligence algorithm and evaluate its performance
Optimize the GANN model using a genetic algorithm and evaluate its performance
Train the Multi-SVM model and evaluate its performance
Compute performance metrics like MSE, FAR, FRR and accuracy rate
Compare the performance Metrics
Draw conclusions based on the results of the comparative analysis

Figure 2: Flowchart of the comparative analysis of Multi SVM, GANN, and OFSCNN model for Dialect Identification in the Bagri Rajasthani Language

The workflow is as follows:

1. Data Collection: Gather audio recordings of the Bagri Rajasthani dialect.
2. Data Preprocessing: Extract MFCCs from the audio recordings, followed by feature normalization.
3. Data Splitting: Divide the dataset into training (80%) and testing (20%) sets.
4. Existing Multi-SVM Model: Train a Multi-SVM model on the training set and evaluate its performance on the testing set.
5. Existing GA-NN Model: Optimize the GA-NN model using a genetic algorithm and assess its performance.

6. Proposed OFSCNN Model: Enhance the OFSCNN model using a swarm intelligence algorithm and evaluate its performance.
7. Performance Metrics: Calculate performance metrics, including Mean Square Error (MSE), False Acceptance Rate (FAR), False Rejection Rate (FRR), and Accuracy Rate, for each model.
8. Comparative Analysis: Conduct a comparative analysis of the performance metrics.
9. Conclusion: Draw conclusions based on the outcomes of the comparative analysis and provide insights for future research directions.

3.5 Algorithms used for dialect recognition system of Rajasthani language:

3.5.1 Proposed Optimized Featured Swarm Convolutional Neural Network (OFSCNN)

The OFS-CNN is a DL model that combines the strengths of CNNs and PSO to enhance the accuracy of speech DR. The OFSCNN algorithm involves the following steps:

1. **Data Preprocessing:** The speech signals are preprocessed to extract features such as MFCCs.
2. **Feature with Optimization:** PSO is used to optimize the set of features used by CNN to improve classification accuracy. The PSO algorithm searches for the optimal set of features that maximize the accuracy of the classification.
3. **Training:** The optimized features are used to train a CNN. It consists of several layers of convolutional PLs that learn to extract parts from speech signals. The weights of the CNN are prepared using a backpropagation algorithm to minimize the classification error.
4. Testing: The trained CNN is used to classify speech signals in the testing dataset. The CNN inputs the pre-processed speech signals and outputs the predicted dialect.

The OFSCNN algorithm has several advantages over traditional approaches to speech dialect recognition. The PSO algorithm optimizes the set of features used by the CNN, improving the classification accuracy. The CNN construction is considered to learn complex components from the speech signals, which allows the model to capture subtle differences between dialects. Additionally, the OFSCNN algorithm is highly automated and can learn from large datasets, reducing the need for manual tuning and improving the approach's scalability.

Algorithm:

Input: Speech Data Samples

Output: DF as dialects, Accuracy, FAR, FRR, MSE.

```
1. Load Input Data DS1, OFSCNN_training_model;
2. Preprocess the Input DS1;
3. Identify Ns (noise signal);
SNR_val (Signal-to-Noise Ratio) =SNR (Ns,Signal_noise);
4. SSignal (Smooth Signal) = smooth (Ns);
5. Divide the signal into various Frames F
6. For each Frame F(i)
```

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

7. Module 1 Feature Extraction Using MFCC

- Apply pre-emphasis filter on speech DS1 which will enhance high-frequency components and reduce the effect of noise.
- Frame the Speech DS1 into overlapping frames of length N and with a hop-size of M.

Pre-emphasis Filter:

Y (n) = x (n) – alpha*x (n-1);

- Apply window Function to each frame to reduce spectral leakage.

W (n) =0.54-0.46*cos (2*pi*n/);

- Evaluate the power spectrum of each frame using the DFT (Discrete Fourier Transformation).
- Apply mel filter bank to the power spectrum to obtain the mel spectrogram.

Hm(k)=

$$\begin{cases} 0 & k < f(m-1) \\ \mathrm{k} - \dfrac{\mathrm{f(m-1)}}{\mathrm{f(m) - f(m-1)),}} & \mathrm{f(m-1) \leq k} < f(m) \\ \dfrac{f(m+1)-k}{f(m+1)} - f(m); & f(m) \leq k < f(m+1) \\ 0 & k \geq f(m+1) \end{cases}$$

Where f (m) is the center frequency of the mth mel filter.

- Evaluate the DCT (Discrete Cosine Transformation) of the log mel spectrogram.

DCT:

x (k) = sum_n = 0^n-1, x(n)*cos(pi*k*(n+0.5/N);

for k= 0,1,…..N-1;

- Logarithm:

Log_Spec = log (mel_spec);

8. Module 2 Optimized Swarm Convolution Neural Network (OSCNN)

Initialize xi, vi, iterations, Pbest, Gbest \\ vi (velocity), xi(position), iterations (feature extracted data).

Generate random particle (p)

For each particle (i)

Evaluate (Feedforward Neural Network) FFN (fi)

Update pbest, gbest

End for

While iteration

For each particle I

```
Update vi, xi
If (xi>limit then xi= limit)
Evaluate FFN fi
//Collect optimized feature vector
Update Pbest, Gbest as TFV (Term Frequency Vector)
                        End if
End for
End while.
  End for.
```

9. Load optimized feature set TFV and simulate

10. Generate classified labels

11. Simulate labels from F as DF //Identified Dialect signal frame

12. Show dialects speech frame DF

13. Compute the performance of all the classified frames vs input //, Accuracy, FAR, FRR, MSE

MSE = $\frac{1}{N}\sum_{j=1}^{n}(yj - y'j)^2$ ……… (i)

Accuracy = Tp + Tn / Tp + Tn + Fp + Fn *100 % ………..(ii)

FAR= Fp / Fp + tn

FRR=Fn / tp + fn

Here, Tp : true positive, Tn: true negative, Fp : false positive, Fn: False negative, MSE: mean square error rate, FAR: false acceptance rate; FRR : false rejection rate.

14. Stop

3.5.2 Existing Genetic Algorithm Neural Network (GANN)

Input: Speech Data Samples

Output: DF as dialects, Accuracy, FAR, FRR, MSE.

1. *Define the parameters of the genetic algorithm*

popul_size = 100

crossr_rate = 0.8

mut_rate = 0.1

max_gen= 100

2. *Define the parameters of the neural network*

inp_size = 13

hid_size = 10

out_size = 26

```
learning_rate = 0.01
3.  Initialize the population of chromosomes randomly, Each chromosome is a vector of
    weights for the neural network
population = random(popul_size, inp_size * hid_size + hid_size * out_size)
4.  Define the fitness function for evaluating each chromosome, The fitness is the accuracy of
    the neural network on the training data
def fitness(chromosome):
5.  Reshape the chromosome into weight matrices for the neural network
W1 = reshape (chromosome [0 : inp_size * hid_size], (inp_size, hid_size))
  W2 = reshape (chromosome [inp_size * hid_size:], (hid_size, out_size))
6.  Initialize the accuracy to zero
accuracy = 0
7.  Loop through the training data
for x, y in train_data:
8.  Forward pass the input through the neural network
   h = sigmoid (W1 * x)
    o = softmax (W2 * h)
9.  Compare the output with the target label
if argmax(o) == argmax(y):
10. Increase the accuracy by one if they match
    accuracy += 1
11. Return the accuracy as the fitness value
return accuracy / len(train_data)
12. Loop until the maximum number of generations is reached or a termination criterion is
    met
for generation in range(max_generations):
13. Evaluate the fitness of each chromosome in the population
fitness_values = [fitness(c) for c in population]
14. Select two parents from the population using roulette wheel selection
 par1 = rws (population, fit_val)
  par2 = rws (population, fit_val)
15. Crossover the parents with a certain probability to produce two offspring
if random() <crossover_rate:
    offsp1, offsp2 = crossover (par1, par2)
```

```
else:
    offsp1, offsp2 = par1, par2
```

16. *Mutate the offspring with a certain probability to introduce some variation*

```
if random() <mutation_rate:
    offsp1 = mutate (offsp1)
if random() <mutation_rate:
    offsp2 = mutate (offsp2)
```

17. *Replace the two worst chromosomes in the population with the offspring* population[argmin(fit_val)] = offsp1

```
Population [argmin(fit_val)] = offsp2
```

18. Print the best fitness and chromosome in the current generation

```
print ("Generation:", generation)
print ("Best fitness:", max(fit_val))
print ("Best chromosome:", population[argmax(fit_val)])
```

3.5.3 Existing Multi Support Vector machine (MSVM)

Input: Speech Data Samples

Output: DF as dialects, Accuracy, FAR, FRR, MSE.

1. Apply some preprocessing steps such as noise reduction, normalization, etc. Compute some features such as MFCCs, pitch, energy, etc. Return a feature vector. Define a function to train a multi SVM classifier

```
def extract_feat(signal):
```

2. *Initialize an MSVM object with some parameters such as kernel, C, gamma, etc. Fit the MSVM to the training data X and labels y. Return the trained MSVM. Define a function to predict the class of a speech signal using multi SVMs*

```
def train_msvm(X, y):
```

3. *Extract features from the signal*

```
def predict(signal):
```

4. *Initialize an empty list to store the predictions of each MSVM*

```
feat = extract_feat(signal)
```

5. *Loop over all the MSVMs*

```
predictions = []
```

6. *Predict the label using the current MSVM*

```
for msvm in msvms:
```

7. *Append the label to the predictions list*

```
label = msvm.predict(feat)
```

```
8.  Return the most frequent label in the predictions list as the final prediction
predictions.append(label)
9.  Load the speech data and labels
return mode(predictions)
10. Split the data into training and testing sets
data, labels = load_data()
11. Initialize an empty list to store the trained MSVMs
X_train,  X_test,  y_train,  y_test = tr_test_split(data, labels)
12. Loop over all the unique labels in the training set
msvms = []
13. Create a new multiple label vector where the current label is 0,1,2… and others are n
for label in unique (y_train):
y_multiple = np.where(y_train == label, 0,1,2,…)
14. Train an MSVM classifier using the training data and the multiple-label vector
msvm = train_msvm(X_train, y_binary)
15. Append the trained MSVM to the msvms list
msvms.append (msvm)
16. Loop over all the testing signals
for signal in X_test:
17. Predict the class of the signal using the multiple SVMs
prediction = predict (signal)
18. Print the prediction and the actual label
print (prediction, signal.label)
```

IV. COMPARATIVE RESULTS

A comparative analysis of OFSCNN, Multi-SVM and GA-NN model for Dialect Identification in the Bagri Rajasthani Language in (Table 2) and (Fig. 3)

4.1 Performance Matrices Used:

Mean square error (MSE), false acceptance rate (FAR), false rejection rate (FRR), and accuracy rate are utilised as performance measures to assess the effectiveness of an OFSCNN, Multi-SVM, and GA-NN-based for the Rajasthani language dialect recognition system.

1. **False acceptance rate (FAR)** is the percentage of incorrectly identified dialects among the speech samples that do not actually belong to that dialect. A lower FAR indicates better performance. To calculate this, we used the following formula:

$$FAR = \frac{Fp}{Fp + Tn}$$

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

2. **False rejection rate (FRR)** is the percentage of correctly identified dialects among the speech samples that are incorrectly classified as a different dialect. A lower FRR indicates better performance. To calculate this, we used the following formula:

$$FRR = \frac{Fn}{Fn + Tp}$$

3. **Accuracy rate** is the percentage of correctly identified dialects among all the speech samples in the testing set. A higher accuracy rate indicates better performance. To calculate this, we used the following formula:

$$Accuracy\ Rate = \left(\frac{Tp + Tn}{Tp + Tn + Fp + Fn}\right) * 100$$

4. **Mean square error (MSE)** is a measure of the average squared difference between the predicted and actual dialect labels for a set of speech samples. A lower MSE indicates better performance. To calculate this, we used the following formula:

$$MSE = \frac{1}{N}\sum_{j=1}^{n}(yj - y'j)^2$$

Where: Here, Tp : true positive, Tn: true negative, Fp : false positive, Fn: False negative, MSE: mean square error rate, FAR: false acceptance rate; FRR : false rejection rate.

These metrics are calculated and reported to assess how well the Rajasthani language dialect detection system is performing using OFSCNN, Multi-SVM and GA-NN (Fig.4).

4.2 Various Performance Matrices:

Various Performance Matrices like MSE, Accuracy, FAR and FRR are used in this work and the Performance Matrices Comparative analysis of speech recognition is as shown in Fig. 3. The Comparative Analysis with different proposed and existing models is shown in Table 2.

Table 2: Comparative Analysis with different proposed and existing models

Models Parameters	GA-NN	Multi-SVM	OFSCNN
MSE	0.054	0.0225	0.015
FAR	0.0945	0.0543	0.0012
FRR	0.0402	0.249	0.0097
Accuracy	80.63	93.45	96.95

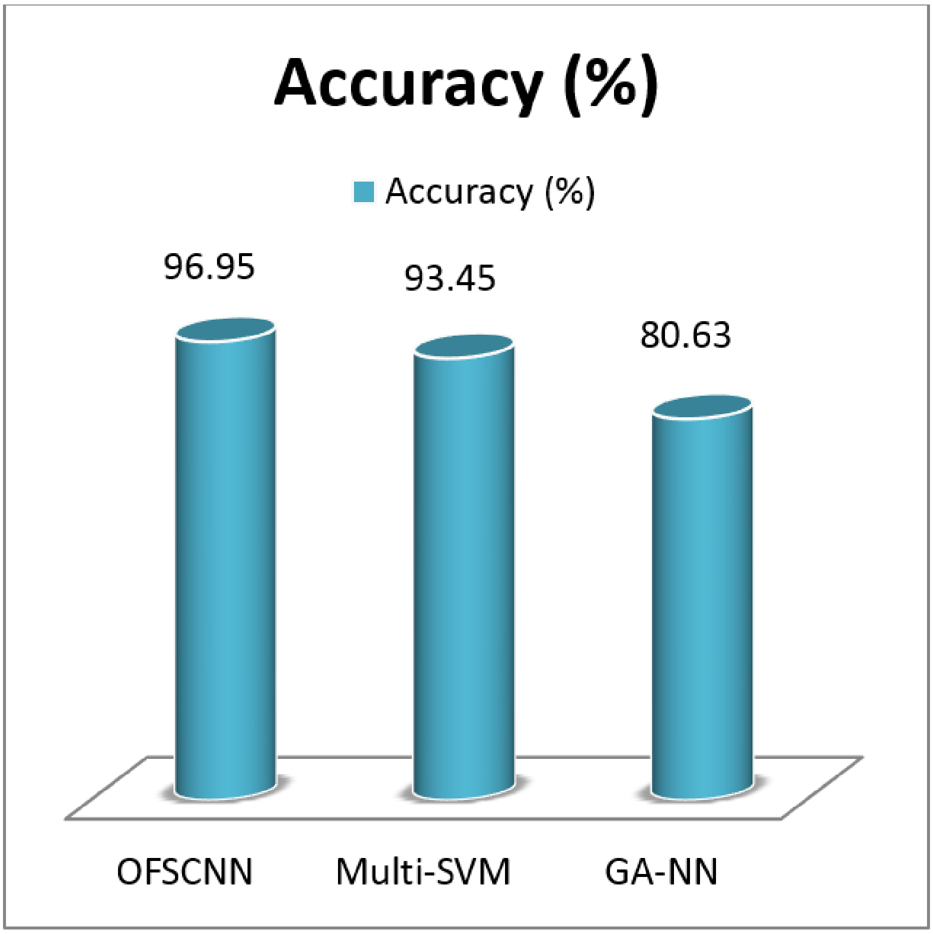

Figure 3: Comparative Analysis of Speech Recognition: Accuracy (%)

4.3 Confusion Matrix:

Error analysis is a crucial step in understanding the behaviour and limitations of a machine learning model. It involves a detailed examination of the errors made by the model on the test data. By understanding the types of mistakes the model is prone to, researchers can make informed decisions about potential improvements or adjustments.

For conducting error analysis, we have included Confusion Matrix which visualizes the model's performance across different classes and analyse the distribution of true positives, true negatives, false positives, and false negatives.

A confusion matrix is a valuable tool that helps assess the performance of my proposed models (OFSCNN, Multi-SVM, and GA-NN) in distinguishing between different dialects. It provides a clear representation of how well my models are classifying instances of the Bagri Rajasthani dialect based on their actual classifications.

Here's how the Confusion Matrix Applies:

True Positives (TP): These are instances where the models correctly predicted the Bagri Rajasthani dialect. In other words, the models correctly identified instances as belonging to the Bagri Rajasthani dialect, and these instances do indeed belong to that dialect.

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

False Positives (FP): These instances are where the models incorrectly predicted the Bagri Rajasthani dialect. The models predicted instances as Bagri Rajasthani, but they were actually not from that dialect.

True Negatives (TN): These are instances where the models correctly predicted that the input is not in the Bagri Rajasthani dialect. The models identified instances as not being Bagri Rajasthani, and these instances are indeed from a different dialect.

False Negatives (FN): These instances are where the models incorrectly predicted that the input is not in the Bagri Rajasthani dialect when it actually is. The models failed to identify instances as Bagri Rajasthani, even though they belong to that dialect.

Table 3: The confusion matrix for Bagri Rajasthani Dataset

	Predicted Positive	**Predicted Negative**
Actual Positive	187	8
Actual Negative	9	258

V. CONCLUSION AND FUTURE SCOPE

In this study, we introduce the OFSCNN model for dialect identification in the Bagri Rajasthani language, alongside the established Multi-SVM and GA-NN models. Our evaluation, conducted on a dataset of audio recordings, reveals intriguing insights into their performance.

The Multi-SVM model exhibits an accuracy of 93.45%, followed by the GA-NN model with an accuracy of 80.63%. The OFSCNN model emerges as the leader, achieving the highest accuracy of 96.95%. These outcomes underscore the potency of both deep learning and machine learning techniques in dialect identification, particularly in the context of low-resource languages.

This research serves as a foundation for extending the proposed method to encompass other Indian languages, facilitating reliable differentiation between diverse dialects. The application of this technology extends to automated translation platforms, natural language processing, and speech recognition.

Although dialect identification for the Bagri Rajasthani language presents inherent challenges, technological progress offers substantial potential for future research and development in this domain. A standardized and extensive corpus can pave the way for a range of speech and language processing applications. The models and algorithms developed herein find utility in domains such as speech recognition, language translation, and language learning platforms. In conclusion, the identification of Bagri Rajasthani language dialects using machine learning and speech processing techniques holds immense promise. As technology continues to evolve, this field of research is poised for significant expansion and advancement in the forthcoming years.

Declaration

Conflicts of interest/ Competing interests: No conflicts of interest that could affect the objectivity or impartiality of this research.

Data Availability Statement: Data is available at

https://github.com/poonamkukana/Speech-Dataset-of-Rajasthani-language.git which contains 500 audio recordings of 50 sentences from 10 speakers of Rajasthani dialect.

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

Funding Sources: This research did not receive any specific grant from any funding agencies.

Author Contributions: Er. Poonam Kukana conceptualized and designed the study, collected data, conducted data analysis, and drafted the manuscript. Dr. Pooja Sharma contributed to literature review, and manuscript editing. Dr. Puneet Sapra and Dr. Neeru Bhardwaj assisted in data interpretation and critically reviewed the manuscript for intellectual content. All authors read and approved the final version of the manuscript and agreed to be accountable for all aspects of the work, ensuring its accuracy and integrity.

References

1) R. K. Aggarwal and M. Dave, "Application of genetically optimized neural networks for hindi speech recognition system," 2011 World Congress on Information and Communication Technologies, Mumbai, India, 2011, pp. 512-517, doi: 10.1109/WICT.2011.6141298.

2) Ali, A., Dehak, N., Cardinal, P., Khurana, S., Yella, S.H., Glass, J., Bell, P., Renals, S. (2016) Automatic Dialect Detection in Arabic Broadcast Speech. Proc. Interspeech 2016, 2934-2938, doi: 10.21437/Interspeech.2016-1297.

3) Q. Zhang and J. H. L. Hansen, "Language/Dialect Recognition Based on Unsupervised Deep Learning," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 26, no. 5, pp. 873-882, May 2018, doi: 10.1109/TASLP.2018.2797420.

4) Zhang, Z., Geiger, J., Pohjalainen, J., Mousa, A. E. D., Jin, W., & Schuller, B. (2018). Deep learning for environmentally robust speech recognition: An overview of recent developments. ACM Transactions on Intelligent Systems and Technology (TIST), 9(5), 1–28.

5) Dogra A., Kaul A., Sharma R.N, Automatic Recognition of Dialects of Himachal Pradesh Using MFCC &GMM, 5th IEEE International Conference on Signal Processing, Computing and Control (ISPCC 2k19), Oct 10-12, 2019, JUIT, Solan, India

6) Themistocleous C.(2019) Dialect Classification From a Single Sonorant Sound Using Deep Neural Networks, https://doi.org/10.3389/fcomm.2019.00064

7) Alotaibi, Y. A., et al. (2019). A canonicalization of distinctive phonetic features to improve arabic speech recognition. Acta Acustica United with Acustica, 105(6), 1269–1277.

8) Kisler T., Winkelmann R., Schiel F.(2019) Styrian dialect classification: comparing and fusing classifiers based on a feature selection using a genetic algorithm, INTERSPEECH 2019 September 15–19, 2019, Graz, Austria

9) Haeb-Umbach, R., et al. (2019). Speech processing for digital home assistants: Combining signal processing with deep-learning techniques. IEEE Signal Processing Magazine, 36(6), 111–124.

10) An, N. N., Thanh, N. Q., & Liu, Y. (2019). Deep CNNs with self-attention for speaker identification. IEEE Access, 7, 85327–85337.

Volume 63 | Issue 02 | February 2024
DOI: 10.5281/zenodo.10695362

11) Cui, X., et al. (2020). Distributed training of deep neural network acoustic models for automatic speech recognition: A comparison of current training strategies. IEEE Signal Processing Magazine, 37(3), 39–49.

12) Kumar, A., & Aggarwal, R. K. (2020). Discriminatively trained continuous Hindi speech recognition using integrated acoustic features and recurrent neural network language modeling. Journal of Intelligent Systems, 30(1), 165–179.

13) Sangwan, P, Deshwal, D, Kumar, D, Bhardwaj, S. Isolated word language identification system with hybrid features from a deep belief network. Int J Commun Syst. 2020; e4418. https://doi.org/10.1002/dac.4418

14) Swamidason, I.T.J., Tatiparthi, S., Arul Xavier, V.M. et al. (2020) Exploration of diverse intelligent approaches in speech recognition systems. Int J Speech Technol (2020). https://doi.org/10.1007/s10772-020-09769-w

15) Chittaragi, Nagaratna B., and Shashidhar G. Koolagudi. "Automatic dialect identification system for Kannada language using single and ensemble SVM algorithms." Language Resources and Evaluation 54.2 (2020): 553-585.

16) Guglani, J., Mishra, A.N (2021). DNN based continuous speech recognition system of Punjabi language on Kaldi toolkit. Int J Speech Technol 24, 41–45 https://doi.org/10.1007/s10772-020-09717-8

17) Sangwan, P, Deshwal, D, Dahiya N., ”Performance of a language identification system using hybrid features and ANN learning algorithms”, Applied Acoustics, Volume 175, 2021, 107815, ISSN 0003-682X, https://doi.org/10.1016/j.apacoust.2020.107815.(https://www.sciencedirect.com/science/article/pii/S0003682X20309208)

18) Thukroo, I.A., Bashir, R. & Giri, K.J. A review into deep learning techniques for spoken language identification. Multimed Tools Appl (2022). https://doi.org/10.1007/s11042-022-13054-0

19) Nanmalar, M., Vijayalakshmi, P. & Nagarajan, T. (2022). Literary and Colloquial Tamil Dialect Identification. Circuits Syst Signal Process (2022). https://doi.org/10.1007/s00034-022-01971-2

20) B. Aarti, S.K. Kopparapu, (2018) Spoken Indian language identification: a review of features and databases. Sādhanā 43(4), 1–14

21) Poonam Kukana (no date) Poonamkukana/speech-dataset-of-rajasthani-language: Contains 500 audio recordings of 50 sentences from 10 speakers of Rajasthani dialect.,GitHub. Available at: https://github.com/poonamkukana/Speech-Dataset-of-Rajasthani-language.git (Accessed: February 13, 2023).

22) Kukana, P., Sharma, P., Bhardwaj, N. Optimized Featured Swarm Convolutional Neural Network (OFSCNN) Model based Dialect Recognition System for Bagri Rajasthani Language. Preprint at https://www.researchgate.net/publication/369726292, DOI: 10.21203/rs.3.rs-2752584/v1 (March 2023)