Accent Recognition

Mukundan, Parth, and Sri Ranga Deep

Problem Statement

"Integration of accent classification into speech recognition systems to accurately identify and adapt to the accents of speakers, improving recognition accuracy and usability, particularly for non-native speakers or those with strong regional accents."

Current voice technologies are primarily based on western and developed datasets.

Improved and fine-tuned speech recognition models are needed to accommodate various accents and ensure inclusive usability.

Literature Survey

S.No	Title	Methods	Features Extracted	Accuracy	Number of Accents	Dataset Size
1	Accent Recognition Using I-Vector, Gaussian Mean Supervector And Gaussian Posterior Probability Supervector For Spontaneous Telephone Speech	SVM, NBC and SRC	Gaussian Mean Supervector (GMS), i-vector and Gaussian Posterior Probability Supervector (GPPS)	56% 50% 58%	5	30,000
2	Identification of the English Accent Spoken in Different Countries by the k-Nearest Neighbor Method	KNN	MFCC	87.3%	6	330
3	Indian Accent Detection using Dynamic Time Warping	Dynamic Time Warping	MFCC	63.4%	4	-
4	Speaker Accent Recognition Using MFCC Feature Extraction and Machine Learning Algorithms	Multi-layer Perceptron, KNN	MFCC	89.1%, 88.2%	7	367*
5	Accent classification using Machine learning and Deep Learning Models	Polynomial SVM, Decision Tree	MFCC	95.434%, 98.054%	5	2140
6	Accent Classification for Speech Recognition	A combination of GMM and SVM	MFCC, First and Second Derivatives of MFCC, Word N-grams, POS N-grams.	82%	-	948
7	English Language Accent Classification and Conversion using Machine Learning	GANs	MFCC, Fundamental Frequency, Aperiodicity	68%	13	-
8	A Machine Learning Approach to Recognize Speakers Region of the United Kingdom from Continuous Speech Based on Accent Classification	KNN, SVM, Random Forest (the accuracy is tested with unscaled, min-max scaled, and standard scaled features)	MFCC	98.4% 97.3% 93.2%	5	17,877
9	Comparison of Feature Extraction for Accent Dependent Thai Speech Recognition System	SVM	Energy Spectral Density (ESD), Power Spectral Density (PSD), Mel-Frequency Cepstral Coefficients (MFCC) and Spectrogram (SPT)	89.3% (M) and 93.8% (F) For MFCC	3	600
10	Accent Classification	SVM, GMM	MFCC, PLP	51.47%	3	10,000

*only number of speakers was divulged, number of samples may vary

Shortcomings found

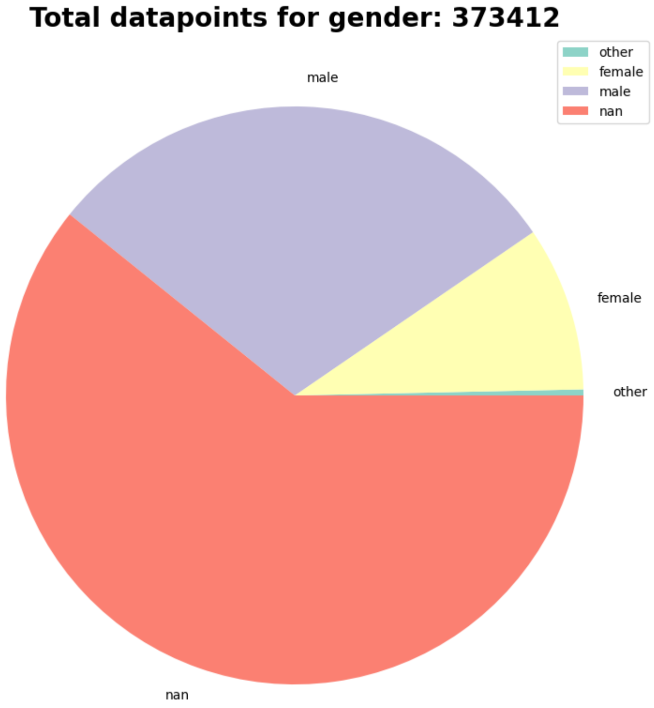
- Most models implemented lacked diversity in their datasets.
- The datasets used were either region specific or were too general, taking a global dataset but taking into account only a handful of accents.
- Some of the papers also operated on very small data sets which caused skewed results.

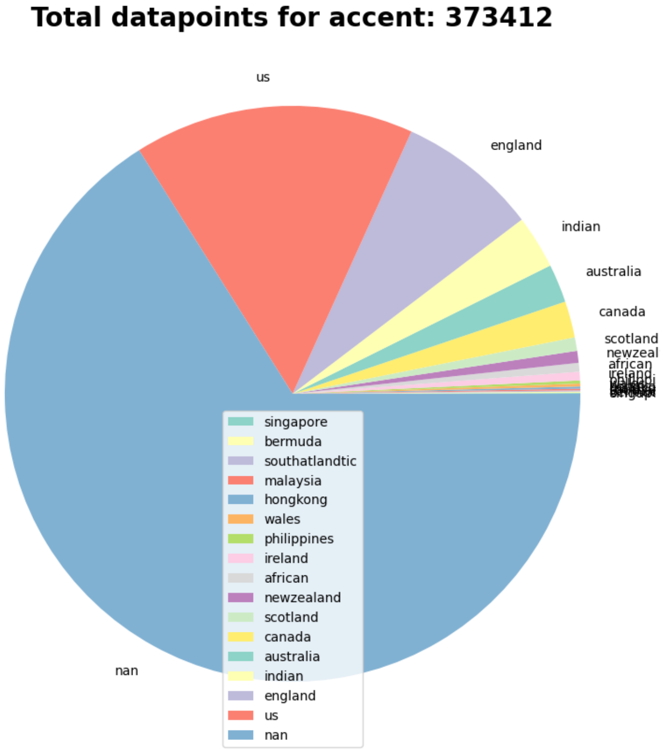
- We are improving on the number of accents classified.
- We are also improving on accuracy.

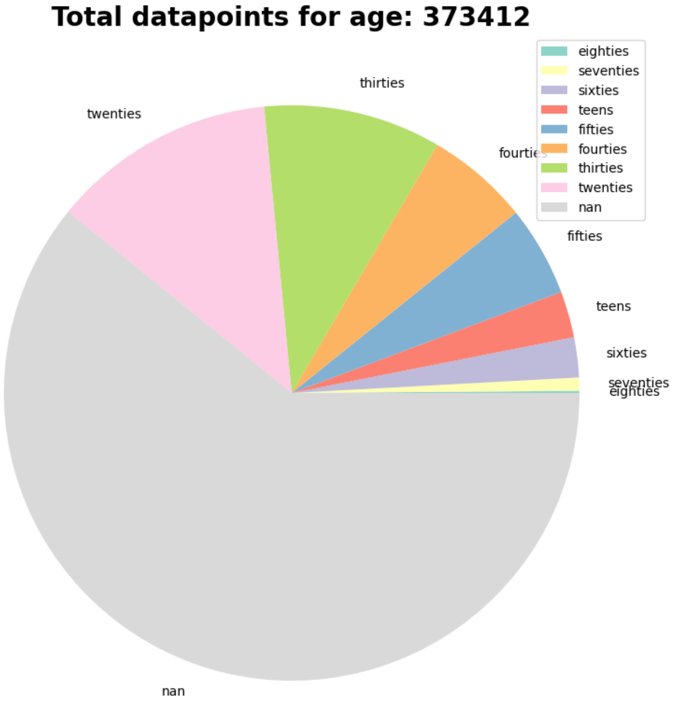
Dataset - Common Voice

- Dataset by mozilla.org
- Audio clips submitted by users (donated)
- Text, self-reported and voted on by users
- Age, gender and accent - self reported

All Data

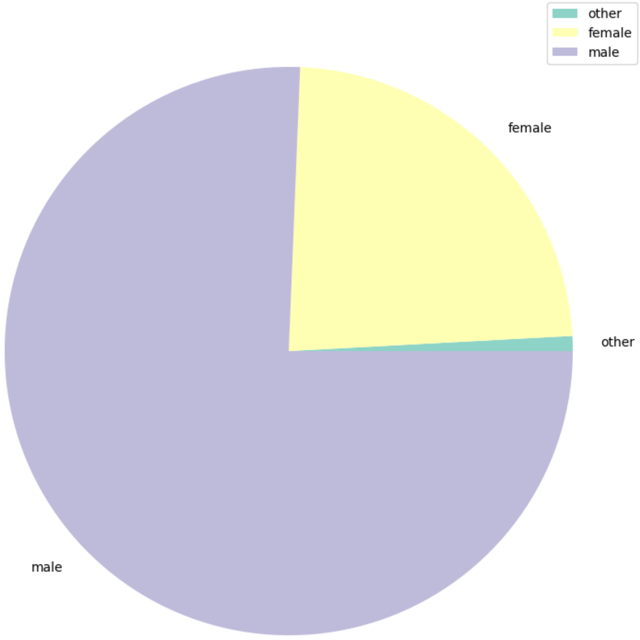

Total datapoints for gender: 373412
other
female
male
nan
male
female
other
nan


Total datapoints for accent: 373412
us
england
indian
australia
canada
scotland
newzeal
african
ireland
nan
singapore
bermuda
southatlandtic
malaysia
hongkong
wales
philippines
ireland
african
newzealand
scotland
canada
australia
indian
england
us
nan

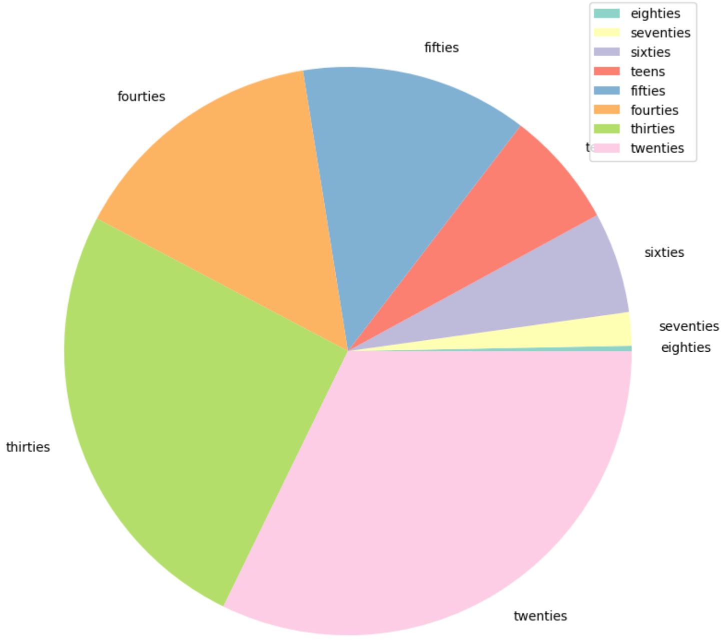

Total datapoints for age: 373412
eighties
seventies
sixties
teens
fifties
fourties
thirties
twenties
nan
thirties
twenties
fourties
fifties
teens
sixties
seventies
eighties
nan

Labeled Data

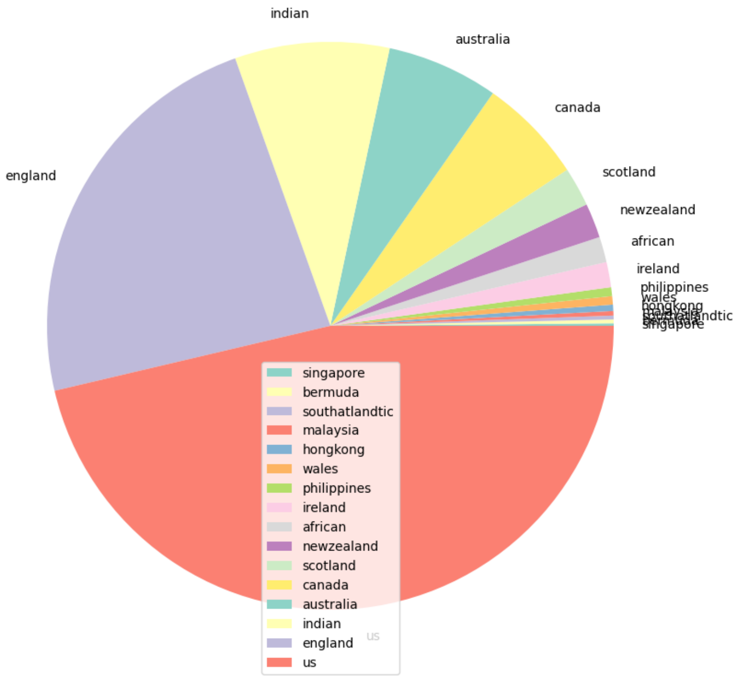
Total datapoints for gender: 146459



Total datapoints for age: 146196



Total datapoints for accent: 126791



Pre-Processing

Data Cleaning and Formatting

- Removing audio above 10s
- Padding audio
- Sampling rate of the audio was already set to 22050 Hz

Pre-Processing

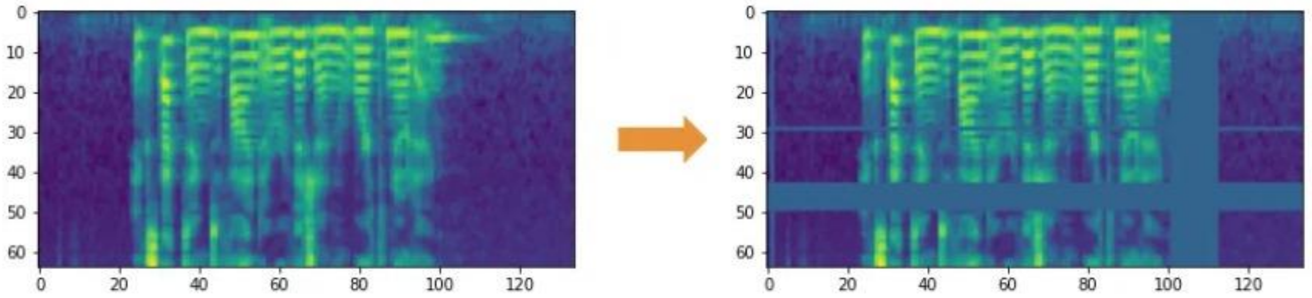
Data Issues

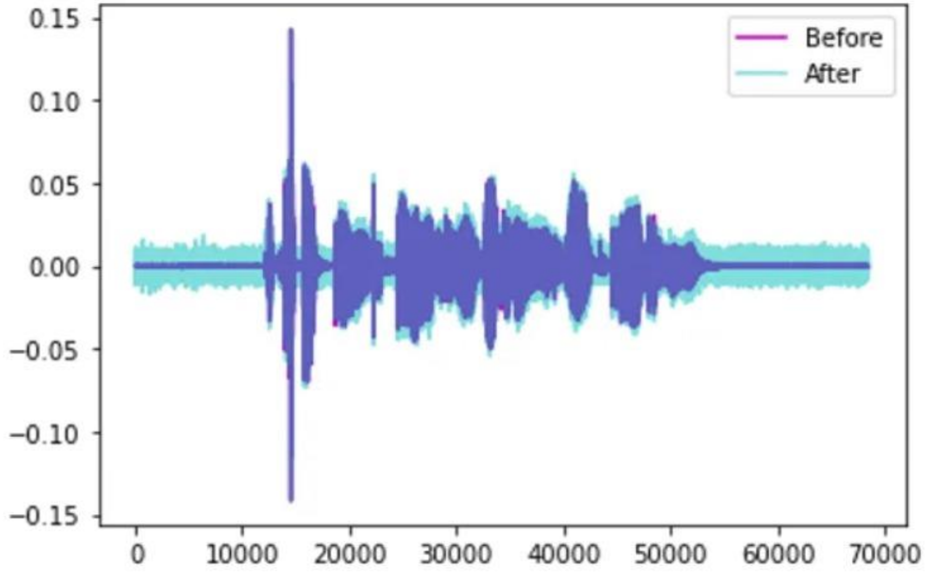
- Imbalance dataset - data augmentation
- Time shift: Shifting the audio to the left or the right by a random amount
- Add noise: Add random values to the sound
- Frequency/Time mask: Removing random frequencies and time bands from the spectrogram.

Not suitable

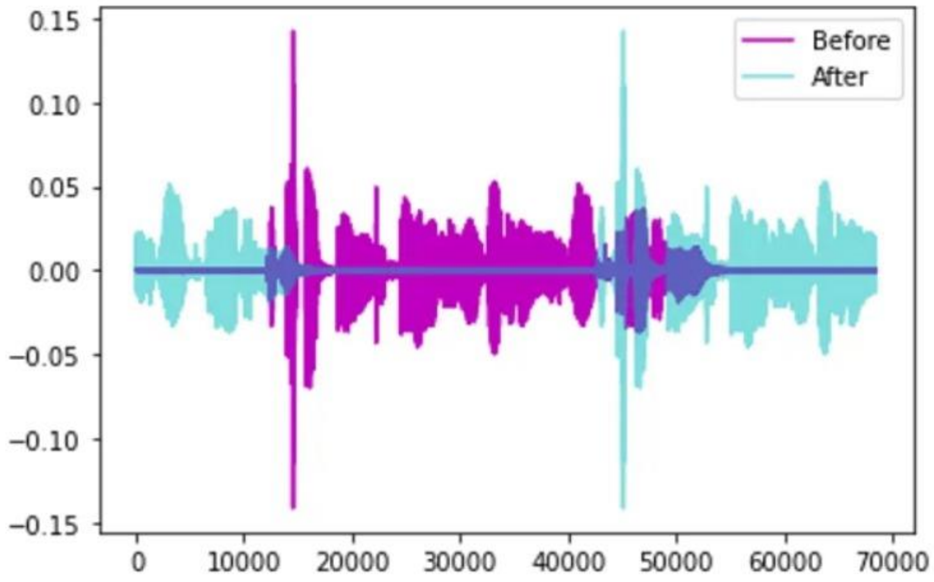
- Pitch Shift: Randomly modifying the frequency of parts of the sound
- Time stretch: Randomly slow or speed up the sound

Data Augmentation





Augmentation by Adding Noise (Image by Author)



Augmentation by Time Shift (Image by Author)

https://towardsdatascience.com/audio-deep-learning-made-simple-part-3-data-preparation-and-augmentation-24c6e1f6b52

Features Extraction

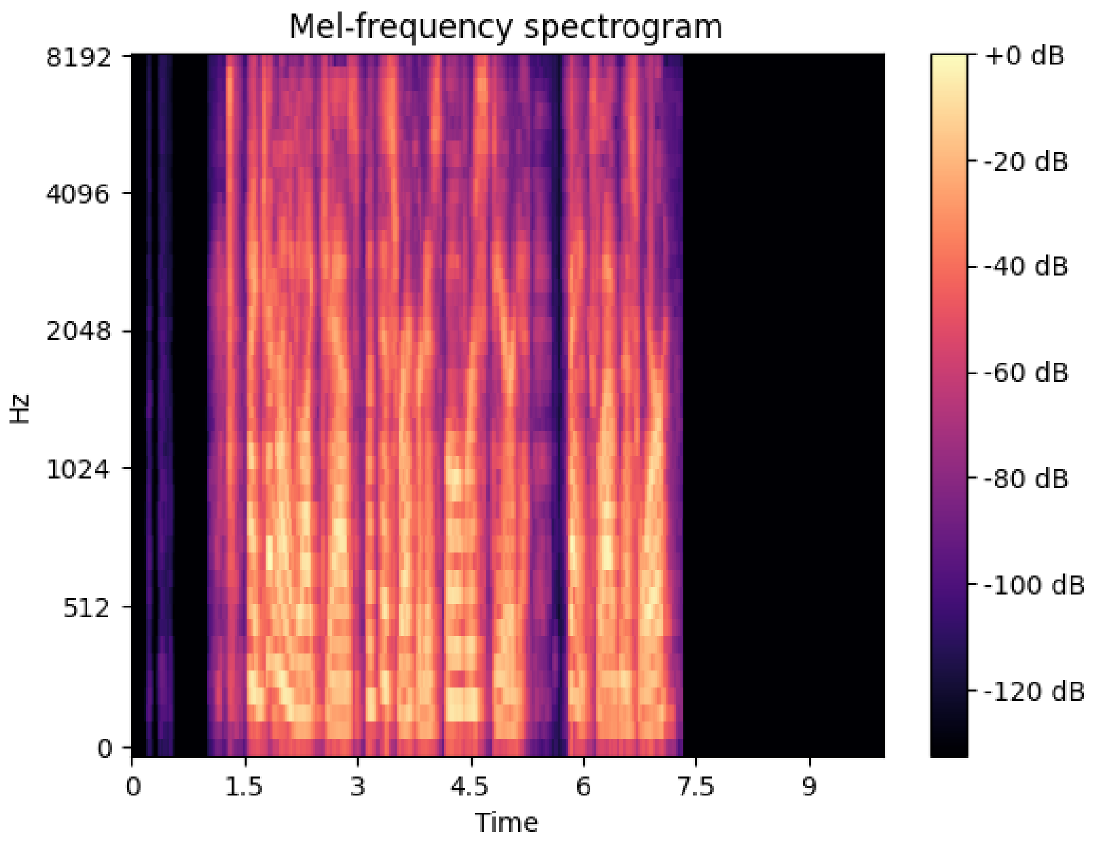
- **Framing and Windowing**: Dividing audio into short time segments and applying a function for analysis.
- **Fast Fourier Transform**: Converting audio from the time domain to the frequency domain to identify frequency components.
- **Mel Filtering**: Applying filters based on human hearing sensitivity to capture important frequency components.
- **Discrete Cosine Transform:** Transforming filterbank energies into coefficients that represent audio features compactly.
- **Obtained MFCCs:** Mel-Frequency Cepstral Coefficients capture essential audio characteristics for tasks like speech recognition

Features Extracted

- **Mel-Frequency Cepstral Coefficients**
- **Time Series MFCC**
- **Zero Crossing Rate**
- **Spectral centroid**
- **Root mean square energy**

Visualization

- Mel scale on the Y axis
- Time on X axis
- Colours represent power in dB



$$Mel(f) = 2595 * log_{10}\left(1 + \frac{f}{700}\right)$$

Methodology

- MFCC, identified as the most prevalent and successful feature in human speech data analysis, is selected as the primary feature extraction method.

- KNN, and SVM, proven effective in accent classification across multiple studies, are chosen as the classification models.

- In our pre-literature study research we also discovered articles implementing neural networks

Methodology

- MFCCs are extracted and standardized and then passed onto to the models.
- Models
  - KNN - A simple algorithm that classifies new data points based on their similarity to the existing data points in the training set.
  - SVM - Another simple but powerful algorithm which finds the hyperplane that separates the classes in the feature space.
  - NN - Due to its architecture it is able to handle complex input data like MFCCs and learn to map them to their classes.

Methodology

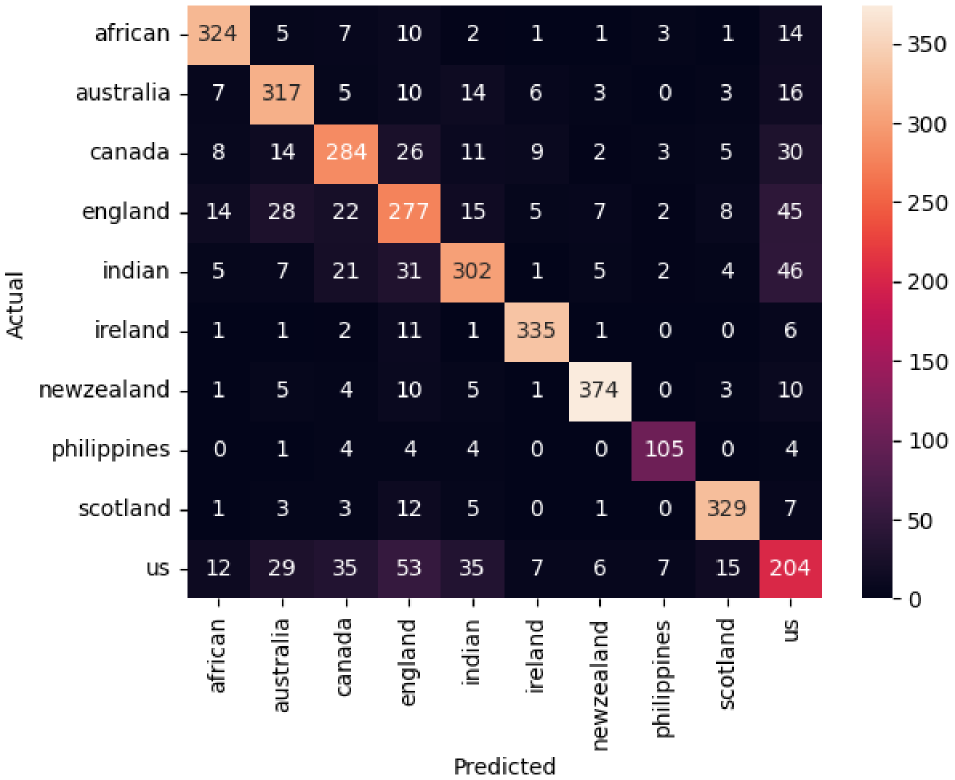
- Challenges faced
  - Hit a roadblock with improving accuracies of our models.
- Solutions tried
  - Used extra Features like zero-crossing rate (dominant frequency), root mean square energy (loudness), pitch, and spectral centroid (tone).
  - Augmentation techniques
    - Adding noise to the audio
    - Stretching the audio
    - Time shifting the audio

Methodology

- Solutions tried
  - LSTM
- Working solutions
  - Used RandomOverSampler, and SMOTE to correct biases in the dataset.
  - Used features clustering methods like DBScan to remove outliers from the dataset.

Support Vector Machine (SVM)

- Used GridSearchCV to find the best hyperparameters.
- Radial Basis Function Kernel with a C of 18.
- Performance Metrics
    - Accuracy - 78.216%
    - Balanced accuracy - 79.235%
    - F1-score - 0.781

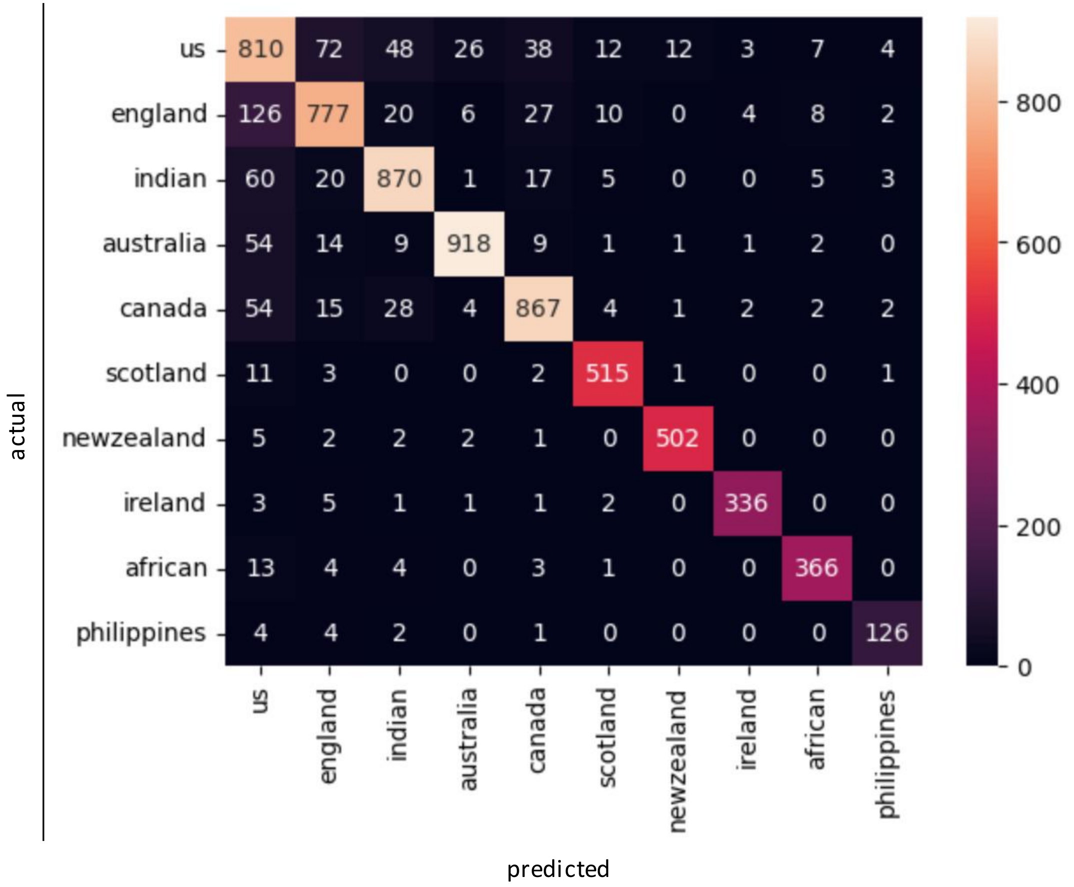


Neural Network (NN)

- Hyperparameters
  - Input Layer
  - Hidden Layer 1: 1024 neurons, activation: relu
  - Dropout Layer: 0.5
  - Hidden Layer 2: 512 neurons, activation: relu
  - Dropout Layer: 0.4
  - Hidden Layer 3: 256 neurons, activation: relu
  - Dropout Layer: 0.2
  - Output Layer: activation: softmax
  - Optimizer: adam, loss: categorical crossentropy, metrics: accuracy and F1 score

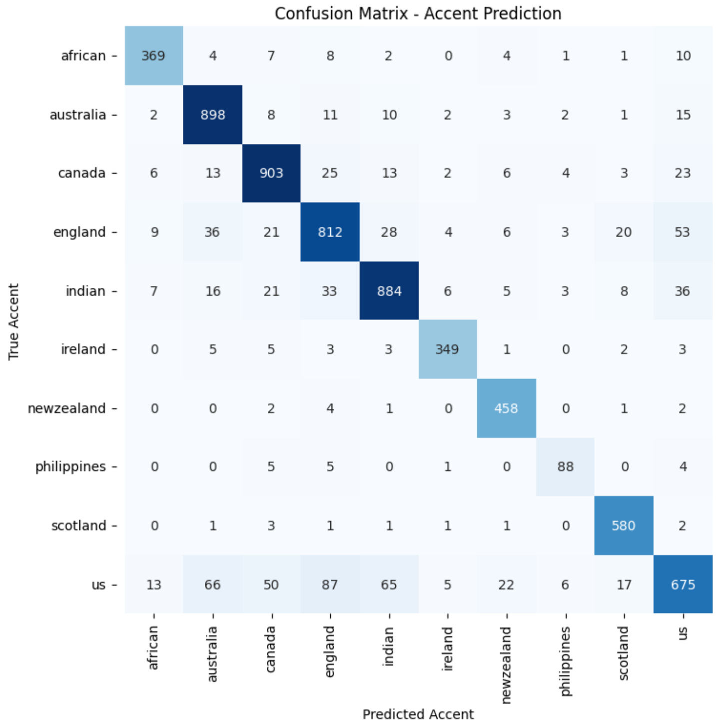
Neural Network (NN)

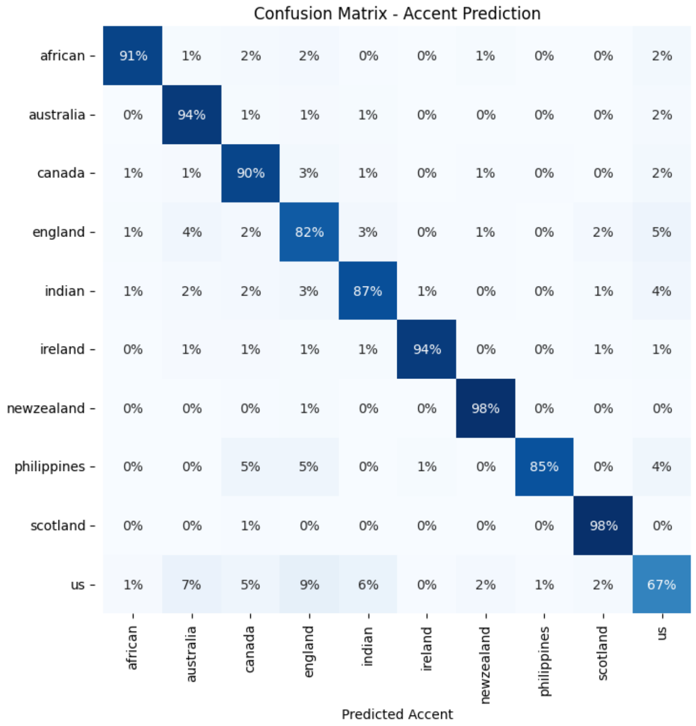
- Performance Metrics
  - Accuracy – 88.15%
  - Balanced Accuracy – 90.21%
  - F1-score – 0.88



K-Nearest Neighbours (KNN)

- Without DBScan and outlier removal
- Performance Metrics
  - Accuracy – 86.38%
  - Balanced Accuracy – 88.50%
  - F1-score – 0.86

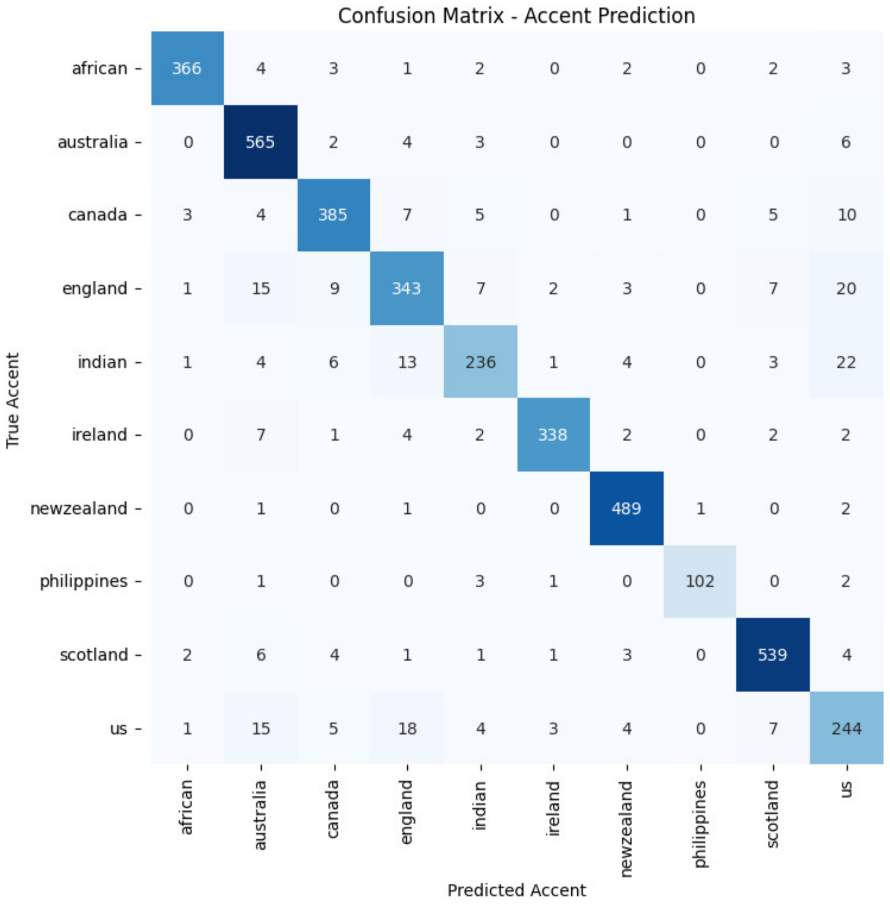


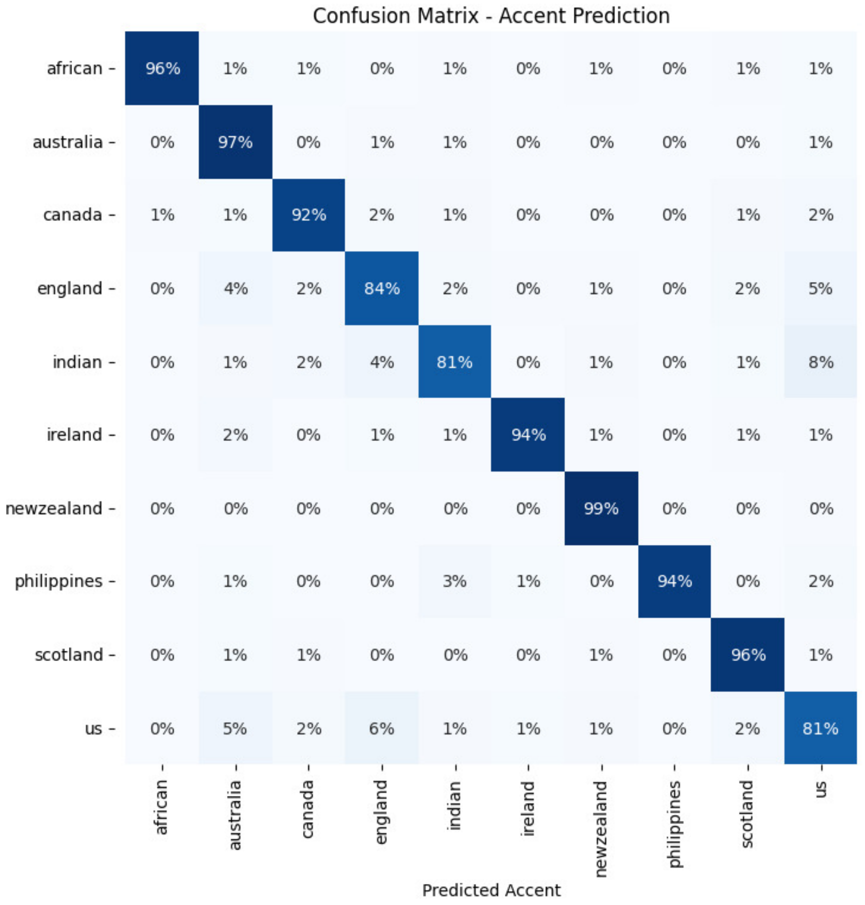


K-Nearest Neighbours (KNN)

- With DBScan and outlier removal
- Hyperparameters for DBScan
  - Epsilon = 2.6
  - Min_samples = 2
- Hyperparameters for KNN
  - Neighbours = 1
  - Distance = Minkowski
- Performance Metrics
  - Accuracy – 92.97%
  - Balanced Accuracy – 92.03%
  - F1-score – 0.92

K-Nearest Neighbours (KNN)





Deployment and Applications

- Adapting to changing accents
- Deployment in smart room/classrooms
- Applicable to Plaksha AI assistant

Thank You!!!