Gender Recognition by Voice

Mohammad Al-Fetyani
https://www.linkedin.com/in/malfetyani/

August 4, 2020

1 Definition

1.1 Problem Statement

The problem states the following:

> This task includes building a machine learning model/solution to detect the gender from speech. In other words, given an utterance that is spoken by any person, detect the gender of the speaker.

Simply, the problem is a classification problem, in which the trained model should take a waveform as its input and output speaker's gender, whether male or female.

The size of the dataset needed to train a good-enough model varies according to the literature. One research by Maka and Dziurzanski [2014] shows that small corpus can be used to build an acceptable gender identification model. The authors used the TIMIT corpus which consists of 630 speakers (438 male and 192 female speakers) with a total length of 36 minutes [Garofolo, 1993]. Another research is conducted by Levitan et al. [2016] where the researchers used the HMIHY ("How May I Help You") corpus that includes 5002 utterances from 1654 speakers, with an average utterance duration of about 6 seconds, and achieved an accuracy of 92.1%.

Another research by Buyukyilmaz and Cibikdiken [2016] where the authors used a dataset with 3168 records of male and female voices to develop a multi layer perceptron model. They managed to achieve an accuracy of 96.74% on the test est. One final research is carried out by Son et al. [2019], the writers analyzed a dataset of 335 speakers (225 male and 117 female) with a total speech length of 370 minutes. They developed an RNN model that achieved overall accuracy of 69.8% on gender recognition.

From aforementioned researches, one can conclude that large datasets are not essential for the task this project tackles. However, I will use the Common Voice dataset, which consists of 195,776 training voice samples and 3,995 testing voice samples, for two reasons. First, it would help build robust and reliable gender classifier. Second, I will utilize deep learning for this task, hence, large dataset achieve better performance.

The problem states that gender must be recognised from speech (waveform). Therefore, an already filtered and cleaned tabular dataset is not an option here. In addition, the chosen dataset must include many speakers from a wide range of regions for better generalization.

1.2 Metrics

This is a supervised problem, hence, the gender of the speakers is provided. This is a classification problem as well. Therefore, accuracy metric can be used to measure the performance of the developed

models. Accuracy is defined as follows:

$$\text{accuracy} = \frac{\text{number of correctly classified samples}}{\text{number of all samples}} \tag{1}$$

Other metrics, like AUC and F1-score, are also applicable and will be considered in this project, although, such metrics are addressed more for imbalanced datasets. One can rely only on accuracy to evaluate the models because it gives an indication of how well the models perform and it is easy to interpret. Additionally, the dataset in hand can be made balanced.

2 Analysis

2.1 Data Exploration

As mentioned earlier, the adopted dataset in this project is the Common Voice corpus, which can be defined as follows:

> Common Voice is a corpus of speech data read by users on the Common Voice website (http://voice.mozilla.org/), and based upon text from a number of public domain sources like user submitted blog posts, old books, movies, and other public speech corpora. Its primary purpose is to enable the training and testing of automatic speech recognition (ASR) systems, but we encourage its use for other purposes as well.

The corpus contains 195,776 training and 3,995 testing voice samples. It includes over 500 hours of speech recordings alongside speaker demographics.

The corpus is split into three parts, which are:

1. The subsets with "valid" in their name are audio clips that have had at least 2 people listen to them, and the majority of those listeners say the audio matches the text
2. The subsets with "invalid" in their name are clips that have had at least 2 listeners, and the majority say the audio does *not* match the clip.
3. All other clips, i.e. those with fewer than 2 votes, or those that have equal valid and invalid votes, have "other" in their name.

.
Each subset of data has a corresponding csv file with the following naming convention:

> "cv-{type}-{group}.csv"

where:
"type" can be {valid, invalid, other},
"group" can be {dev, train, test} except for invalid set.

Each row of a csv file represents a single audio clip, and contains the following information:

- filename - relative path of the audio file
- text - supposed transcription of the audio
- up_votes - number of people who said audio matches the text
- down_votes - number of people who said audio does not match text

- age - age of the speaker, if the speaker reported it
- gender - gender of the speaker, if the speaker reported it
- accent - accent of the speaker, if the speaker reported it

The audio clips for each subset are stored as mp3 files in folders with the same naming conventions as it's corresponding csv file. So, for instance, all audio data from the valid train set will be kept in the folder "cv-valid-train" alongside the "cv-valid-train.csv" metadata file.

2.2 Exploratory Visualization

The corpus include loads of missing values as presented in Table 1. Gender is the only column we interested in, thus, 121,717 samples will be deleted from the dataset. Moreover, gender column is imbalanced as presented in Figure 1 with 55,029 males and 18,249 females.

Column	Number of missing values
age	122008
gender	121717
accent	131065
duration	195776

Table 1: Number of missing values in the corpus.

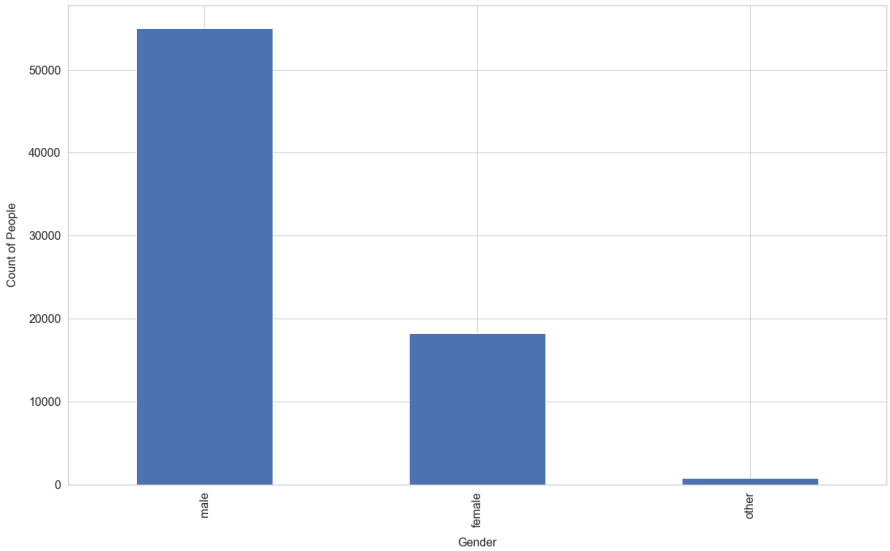

Figure 1: Gender distribution in the corpus.

The age distribution of the speaker is presented in Figure 2. From the figure, it is obvious that the age of the speakers is distributed in all possible ages from teenagers to eighties. Speakers in the twenties are the most in the corpus, with a total of 23,003 speakers, while speakers in the eighties are the least, with a total of 239 speakers.

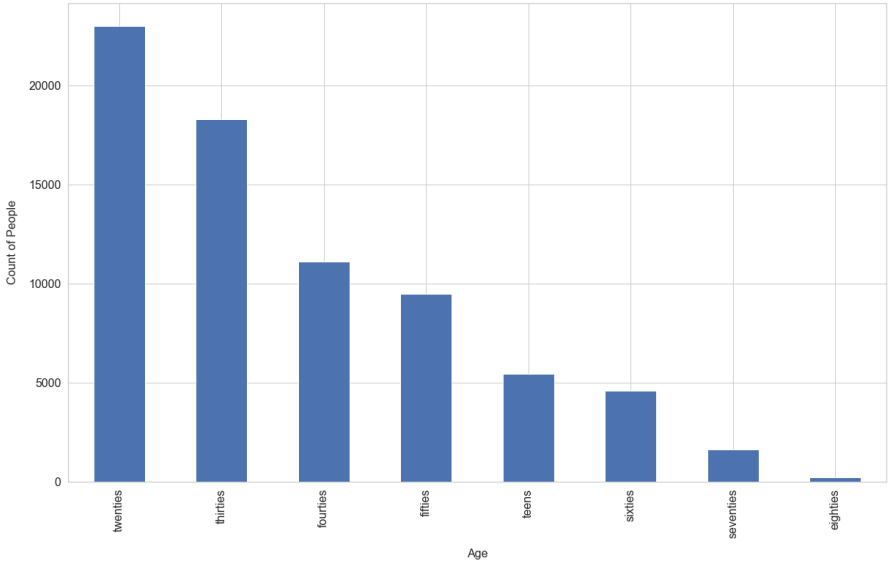

Figure 2: Age distribution in the corpus.

The speakers speak a variety of accent as shown in Figure 3. There are 16 different English accent in the corpus with US and England are the most dominant countries and a few speakers from South Atlantic, Hong Kong, and Singapore.

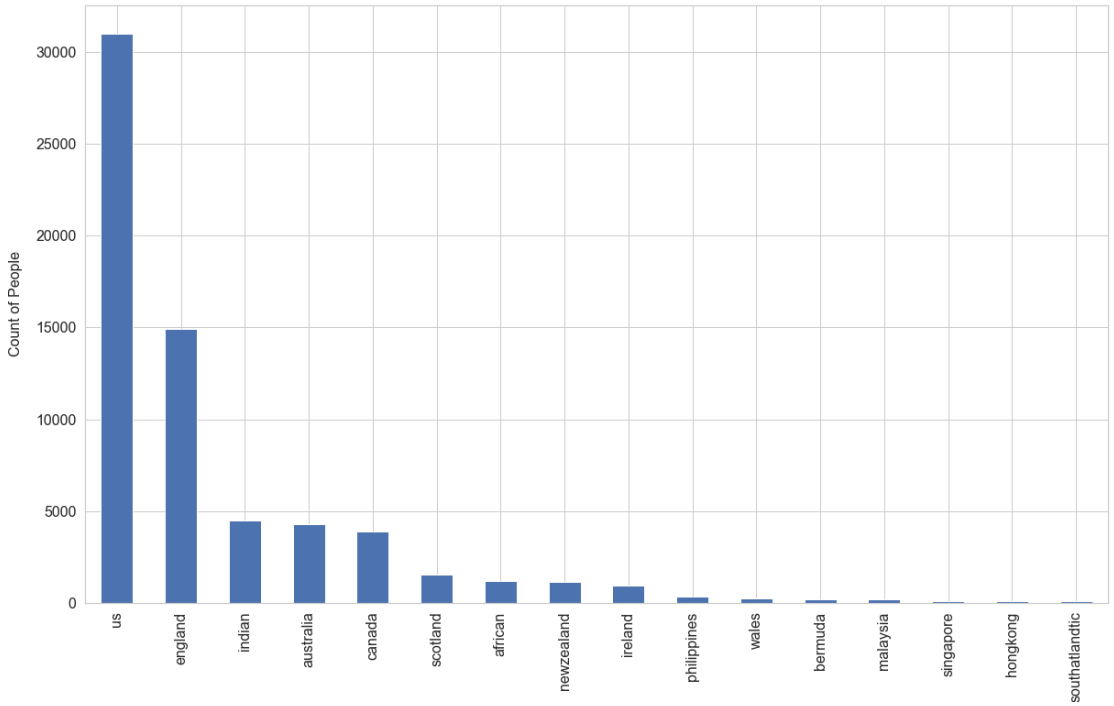

Figure 3: Accent distribution in the corpus.

The lengths of the utterances varies from 0.9s to 10s, without considering outliers, as presented in Figure 4. The have a mean value of 4s and a standard deviation of 2.63s. The upper quantile (75%) has a value of 4.9s, however, I considered values above 10s as outliers. As a result, 315 outliers were found in the

corpus, with a maximum value of 393s.

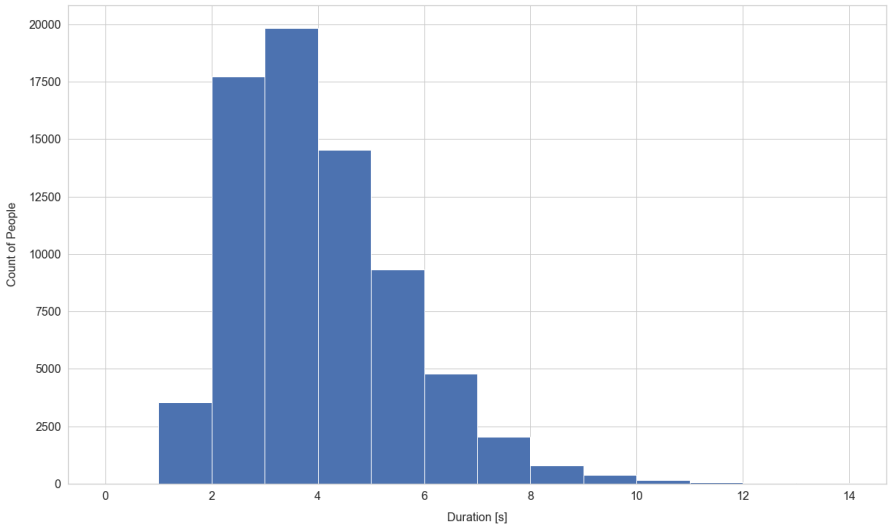

Figure 4: Duration distribution in the corpus.

In conclusion, the analysis confirms that the corpus includes many speakers from different regions of the world, which is necessary for this task, as we mentioned earlier. In addition, the corpus includes many hours of speech, even when missing samples are dropped.

2.3 Algorithms and Techniques

The techniques used in this project begin with the extraction of features from all sound samples, which usually takes about 10 hours. Once the features are extracted, we can store them locally for faster retrieval. Then, we build multiple models, tune them, and choose the best performing model.

Numerous algorithms can be used for classification, and each may perform differently depending on the problem. Therefore, I decided to try 25 different classifiers, choose top 2 performing classifiers and do hyper-parameter tuning on them. These classifiers are:

- GaussianNB
- KNeighborsClassifier
- NearestCentroid
- LinearDiscriminantAnalysis
- QuadraticDiscriminantAnalysis
- LGBMClassifier
- Perceptron
- RidgeClassifierCV
- BaggingClassifier
- AdaBoostClassifier
- RandomForestClassifier
- PassiveAggressiveClassifier
- DummyClassifier
- LogisticRegression
- RidgeClassifier
- CheckingClassifier
- SGDClassifier
- LinearSVC
- DecisionTreeClassifier
- ExtraTreeClassifier
- SVC
- ExtraTreesClassifier
- BernoulliNB
- XGBClassifier
- CalibratedClassifierCV

A deep network of fully connected layers is further developed after many trials, where the initial model was adopted from Buyukyilmaz and Cibikdiken [2016]. The final architecture of the network is presented

in Figure 5. The network consists of an input layer of size 187, followed by 3 hidden layers of sizes 256, 128, and 64 respectively. All hidden layers are followed by a *relu* activation function. Finally, an output layer of size 1 followed by a sigmoid function is used to get the probability of gender being **male**. Moreover, a dropout of 0.3 is used on all hidden layers to reduce overfitting.

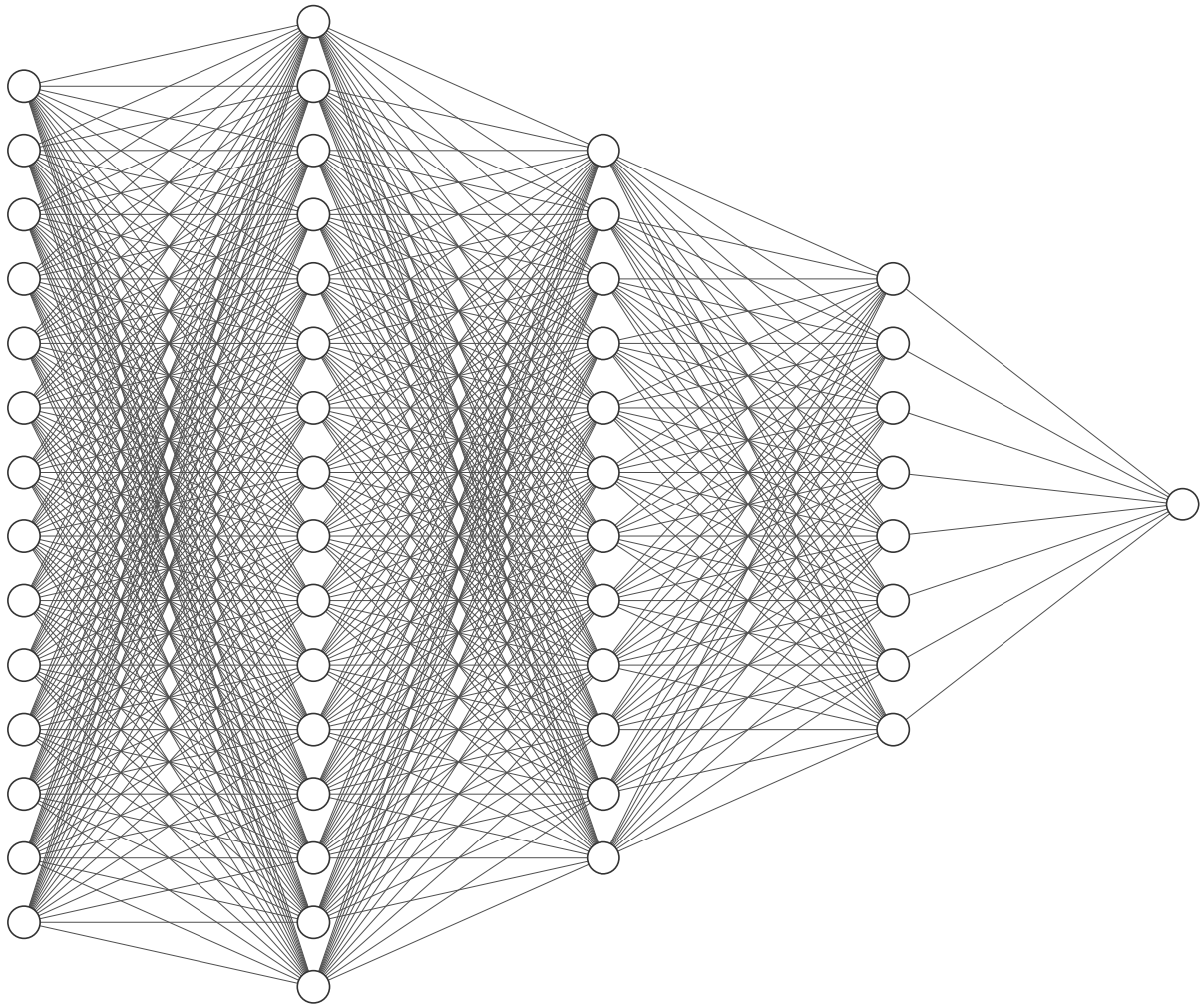

Figure 5: Neural network architecture.

To build a gender classifier using the Common Voice dataset, I followed the following procedure:

1. Filtered the dataset where I kept only male and female, and added the duration of each audio sample to the dataset.
2. Extracted features from all samples. For each sample, I took the mean of its features, stacked all its features, and stored them locally. The features are:
    - MEL Spectrogram Frequency
    - MFCCs
    - Chroma
    - Contrast
3. Divided the dataset into train and validation splits.

4. Applied standard scalar on the train set, and transformed the validation set with the parameters obtained.

5. Balanced the train set by omitting samples of male (Only for the neural network).

6. Fitted all discussed models, and used the validation set to measure their performance.

7. Evaluated all models using a test set because the validation set is used to obtain the best model (seen data).

2.4 Benchmark

A simple classifier, like logistic regression, can be used as the benchmark for this project because it is simple and fast to train. It also gives better results than random guessing for this problem.

Because we can calculate the accuracy of the models, we will also aim for an accuracy over 95%. This is a secondary benchmark to evaluate the results of this project.

3 Methodology

3.1 Data Preprocessing

The feature extraction process took approximately 11 hours on my personal laptop (i7-4980HQ CPU). After dropped all *NaN* and *other* gender type, the dataset became of size 73,278 (55,029 male and 18,249 female). Before training the neural network, I balanced the dataset, which eventually became of size 36,498 (18,249 male and 18,249 female).

This is followed by fitting a standard scalar to the dataset, where the mean of the dataset becomes zero with a standard deviation of 1. A 0.1 split of the train set is used as a validation set to evaluate the models.

3.2 Implementation

This project is programmed completely in Python. The 25 models are implemented with their default hyper-parameters using the lazypredict library (https://github.com/shankarpandala/lazypredict), which facilitate the training process. The neural network is developed completely using PyTorch library.

For the neural network, a binary cross entropy loss function is used since it is a classification problem, and adam optimized is utilized with 1×10^{-3} learning rate. Also, a scheduler that decreases the learning rate with multiple of 0.1 every 10 epochs is used. The training is run for 20 epochs with a batch size of 20.

3.3 Refinement

The initial performance metrics for the 25 models evaluated on the validation set is presented in Table 2, note that the table is sorted in descending order in terms of the balanced accuracy. It is obvious from the table that we managed to achieve really high accuracy of 98% by XGBoost classifier and 97% by LightGBM classifier along with KNN classifier.

The table also indicates that there is a limited space for improvements, so hyper-parameter tuning might not be worth it. Additionally, neural networks are not needed for this task, as the initial performance is already close to optimal.

Model	Accuracy	Balanced Accuracy	ROC AUC	F1 Score	Time Taken
XGBClassifier	0.98	0.97	0.97	0.98	85.69
LGBMClassifier	0.97	0.96	0.96	0.97	7.79
KNeighborsClassifier	0.97	0.95	0.95	0.97	315.04
SVC	0.96	0.94	0.94	0.96	518.04
BaggingClassifier	0.95	0.94	0.94	0.95	356.74
RandomForestClassifier	0.96	0.93	0.93	0.96	191.91
ExtraTreesClassifier	0.95	0.91	0.91	0.94	45.55
DecisionTreeClassifier	0.92	0.89	0.89	0.92	62.79
LinearSVC	0.93	0.89	0.89	0.93	59.37
LogisticRegression	0.93	0.89	0.89	0.93	2.51
SGDClassifier	0.92	0.89	0.89	0.92	3.51
Perceptron	0.91	0.89	0.89	0.91	1.54
LinearDiscriminantAnalysis	0.92	0.88	0.88	0.92	5.00
AdaBoostClassifier	0.92	0.88	0.88	0.91	136.67
RidgeClassifier	0.92	0.87	0.87	0.92	1.51
RidgeClassifierCV	0.92	0.87	0.87	0.92	2.82
CalibratedClassifierCV	0.92	0.86	0.86	0.92	212.46
PassiveAggressiveClassifier	0.90	0.86	0.86	0.90	1.43
NearestCentroid	0.84	0.81	0.81	0.84	1.24
BernoulliNB	0.81	0.79	0.79	0.82	1.46
ExtraTreeClassifier	0.84	0.79	0.79	0.84	1.60
GaussianNB	0.80	0.68	0.68	0.79	1.96
QuadraticDiscriminantAnalysis	0.80	0.66	0.66	0.78	2.00
CheckingClassifier	0.25	0.50	0.50	0.10	1.14
DummyClassifier	0.62	0.49	0.49	0.62	1.19

Table 2: Initial accuracy of the models.

For illustration purposes, however, a neural network is developed and trained. The same neural network that is presented in [Buyukyilmaz and Cibikdiken, 2016] achieved an accuracy of 94% on the validation set. But I managed to raise the accuracy to 96% with the architecture presented in Figure 5, after hard work of hyper-parameter tuning. As a result, XGBoost classifier has been selected as the best performing model.

4 Results

The final model (XGBoost classifier) is evaluated on the test set, which was left aside during training. The parameters of the final model are the default parameters and can be found here. The performance metrics of the model are presented in Table 3. Since the test set is unseen dataset, this means that the final model can achieve well on unseen datasets with almost perfect accuracy.

Metric	Train set	Validation set	Test set
Accuracy (%)	99.9	97.9	97.1
Balanced Accuracy (%)	99.9	97.0	95.8
F1 Score (%)	99.9	98.6	98.0
ROC AUC (%)	99.9	97.0	95.8

Table 3: Performance metrics of the final model.

Although the performance on the train set is near perfect, there is no clear sign of overfitting. Figure 6 presents the learning curve during training, each iteration represent an epoch. The error of both sets is constantly decreasing, which indicates an excellent training process.

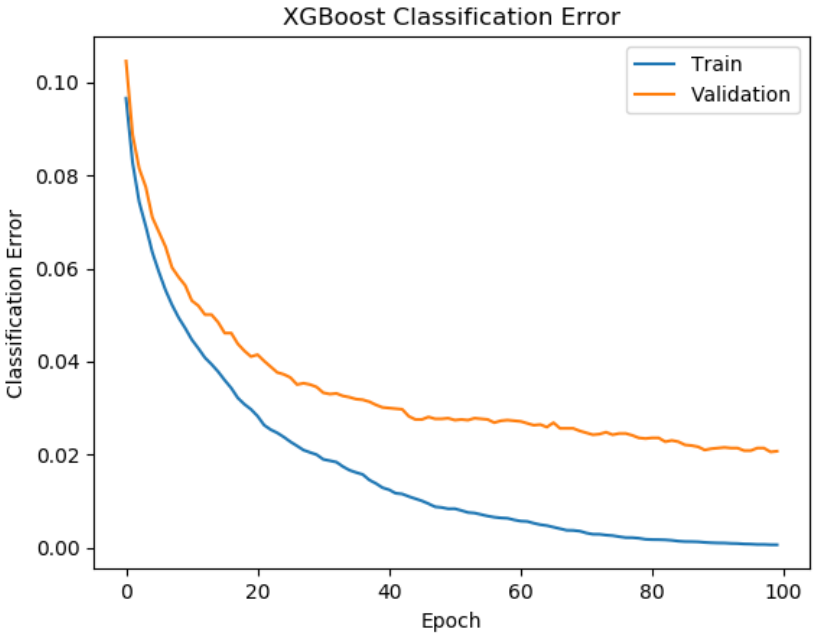

Figure 6: Training learning curve of the final model.

The model's almost perfect score can be due to a clear difference between male and female audio features, making the dataset well separable. A good model also has an impact on the score as we have seen from Table 2.

It is to be noted that the neural network model accuracy on the test set was found to be 90%, which is far from the final model accuracy.

References

Mucahit Buyukyilmaz and Ali Osman Cibikdiken. Voice gender recognition using deep learning. In *2016 International Conference on Modeling, Simulation and Optimization Technologies and Applications (MSOTA2016)*. Atlantis Press, 2016.

John S Garofolo. Timit acoustic phonetic continuous speech corpus. *Linguistic Data Consortium, 1993*, 1993.

Sarah Ita Levitan, Taniya Mishra, and Srinivas Bangalore. Automatic identification of gender from speech. In *Proceeding of speech prosody*, pages 84–88, 2016.

Tomasz Maka and Piotr Dziurzanski. An analysis of the influence of acoustical adverse conditions on speaker gender identification. In *XXII Annual Pacific Voice Conference (PVC)*, pages 1–4. IEEE, 2014.

Guiyoung Son, Soonil Kwon, and Neungsoo Park. Gender classification based on the non-lexical cues of emergency calls with recurrent neural networks (rnn). *Symmetry*, 11(4):525, 2019.