Classification of Vowel and Gender using Deep Learning

Dominic Nzimi

Abstract

The motivation of this coursework will be to explore the use of ML and DL techniques to predict on the Vocal dataset. This paper will present models for prediction of both gender and vowel classification of the audio files. Applications of these models and techniques could be used on single note analyses of the vocals and can be used to get the corresponding voice to text translation and can be particularly useful in improving existing systems particularly in songs to obtain the song lyrics where sounds vary in the duration and frequency spoken.

Modern audio classification uses deep learning techniques which reduces the requirement of musical knowledge which was previously required for designing good features. In many ways, the previous research methods that were used can help us better understand and speculate on the inner workings of some of the Deep Learning algorithms.

Dataset

The dataset that is used is the Vocal17 dataset available on Jhub [1]. Access to the original larger dataset can be obtained from [2] linked below in the reference. Dataset contains a total of 11 female singer and 11 male singer each of which contains different techniques of audio. The audio dataset was down sampled to 22050Hz before it is used in the network. The enables improvement in training by improving the speed when training deeper neural networks.

Input data of Mel-spectrogram images are produced from the audio datasets to produce a 2D image to be used for training ML models for audio and enables us to use 2D Convolutional layers.

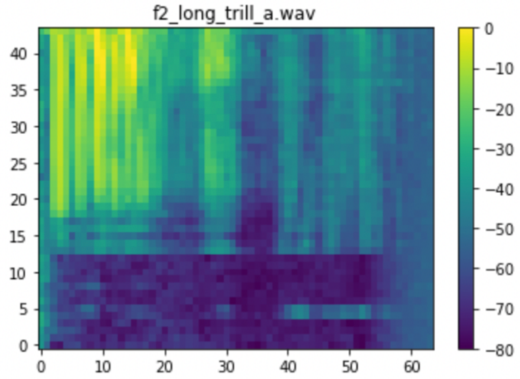
Data preparation

The input data was initially pre-processed by converting the original WAV file into a Mel-Spectrogram using touch audio Mel-Spectrogram function. Librosa library could have also been used to achieve the same task. The initial inputs for our Mel spectrogram used were

- Sample rate = 22050
- Number of samples = 22050
- Hop length = 512
- Number of Mel filter banks = 64
- Size of fft = 1024

Due to computational limitations of Jhub, only this subset of data could be used due to memory limitations. The ‘vocal by vowel’ dataset which had all the files listed in a single directory was used. This required creating a csv Dataframe with the file names for loading the audio files and the target labels. Mel-Spectrogram transformation on input audio files was applied and corresponding pre-processed labels were attached to final data input. A DataLoader was used to increase the speed of implementation and to make use of existing functions and techniques recommended on the official PyTorch tutorials for producing a custom dataset. It also made the code much cleaner to work with by creation of different modules which enabled for more experimentation.

The final dataset size for training/validation/test used was 0.64:0.16:0.2. Some datasets were removed since correct labels did not exist or could not be produced for specific audio files such as excerpts of ‘straight’, ‘spoken ‘and ‘vibrato’.



Example of extracted Mel-Spectrogram for female 2 with pitch ‘long trill’ and ‘vowel A’ audio file. Dataset size for training = 2203, validation = 551, test = 688. It was critical to manually set the seed value when using ‘torch.utils.data.random_split’ function to ensure our experiments are reproducible.

Networks

Network A summary

Network A used is a Connected CNN layer with 3 CNN layers of expanding size which have the same additional features of kernel size = 3, stride =1 and padding = 2. The outputs are then passed through a Flatten layer before being passed through 2 Linear layers the first of which contains a Dropout =0.1. The final size of the second Linear Layer a single dimension number. This output is passed through a Sigmoid function for us to obtain the desired classes.

Network B summary

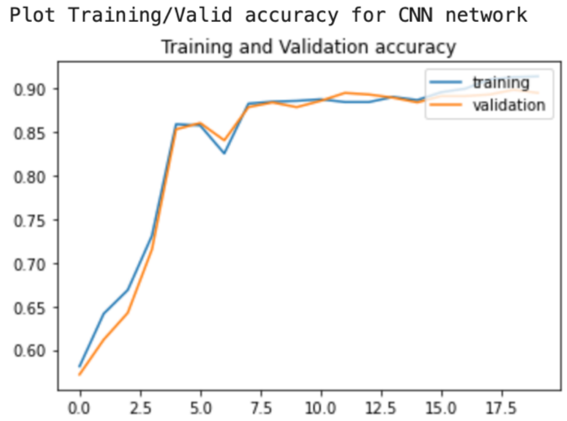
Network B in comparison to Network A is a much simpler model. The input data is first flattened using a Flatten layer before being passed through the Linear layers. 4 fully connected layers are used with dropout layers with dropout rate = 0.1 in-between layers to ensure that our model does not overfit and is able to generalise well to new data.

Task 1: gender classification

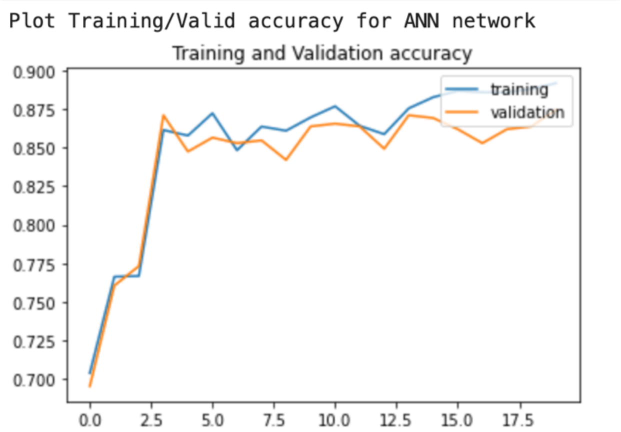
The initial audio data is transformed into a Mel- spectrogram image and loaded into the Data-loader. For the target data, this required extraction of gender from the filename of either m=male or f= female. This was then converted into a suitable format mapping for the training the gender classification model, with the target label is 1=male, 0=female.
After pre-processing, we have a total of 50% for each of the 2 classes. Data was split into Training/Val/Test data. Where both the training and validation accounted for 80% and inside this a further split of 80:20. The validation dataset was used to fine-tune our model.

Since it’s a binary classification, we use binary cross entropy loss function and use the sigmoid function on the last layer to produce an output dimension of 1. The output will correspond to a probability between 0 and 1. We also train using Adam optimizer for 20 epochs due to computational limitations. We also set a default threshold value of 0.5 to predict whether a given probability is either of our classes when evaluating the outputs.

Below is the training and validation accuracy for both networks A and B. We observe that both models reach a training limit where the accuracy does not increase after a certain stage and begins to level out.

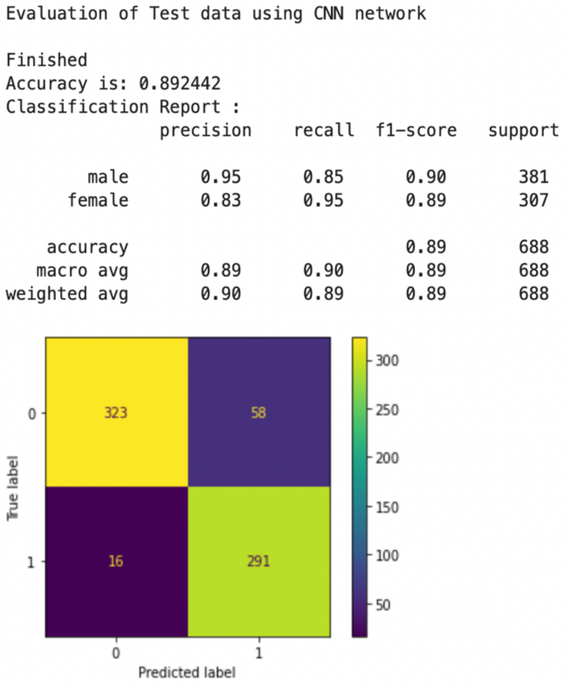


```
Plot Training/Valid accuracy for CNN network
```

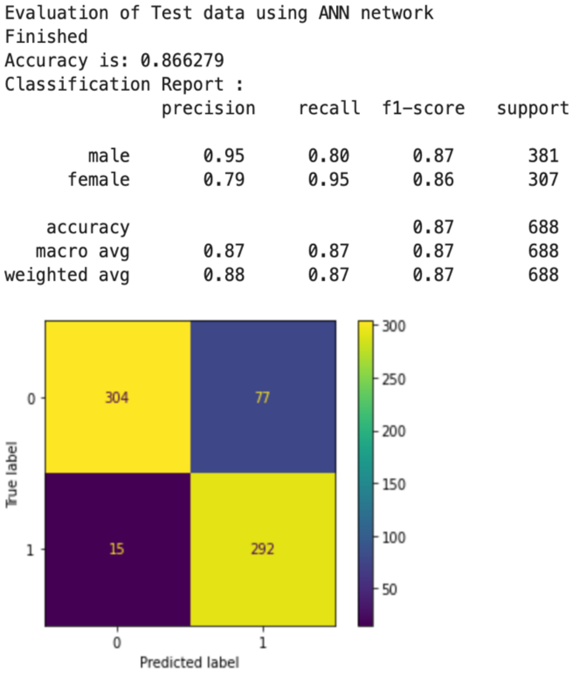


```
Plot Training/Valid accuracy for ANN network
```

Below is the classification report and a confusion matrix plot for both models on the test data. Comparison shows that the Network A performs better than Network B with an accuracy of 89% compared to 86%. We also observe that Network A slightly outperforms Network B when we compare the F1-scores of both models for individual classes.



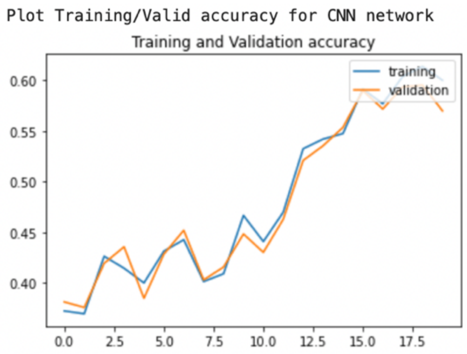
```
Evaluation of Test data using CNN network

Finished
Accuracy is: 0.892442
Classification Report :
              precision    recall  f1-score   support

        male       0.95      0.85      0.90       381
      female       0.83      0.95      0.89       307

    accuracy                           0.89       688
   macro avg       0.89      0.90      0.89       688
weighted avg       0.90      0.89      0.89       688
```

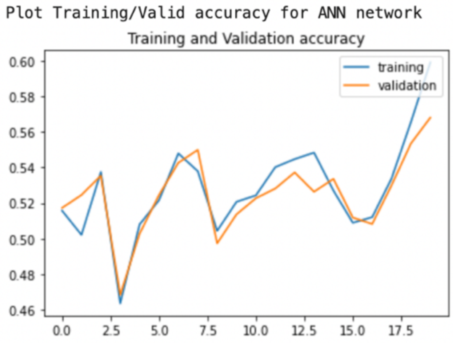


```
Evaluation of Test data using ANN network
Finished
Accuracy is: 0.866279
Classification Report :
              precision    recall  f1-score   support

        male       0.95      0.80      0.87       381
      female       0.79      0.95      0.86       307

    accuracy                           0.87       688
   macro avg       0.87      0.87      0.87       688
weighted avg       0.88      0.87      0.87       688
```

Task2: Classification of Vowel

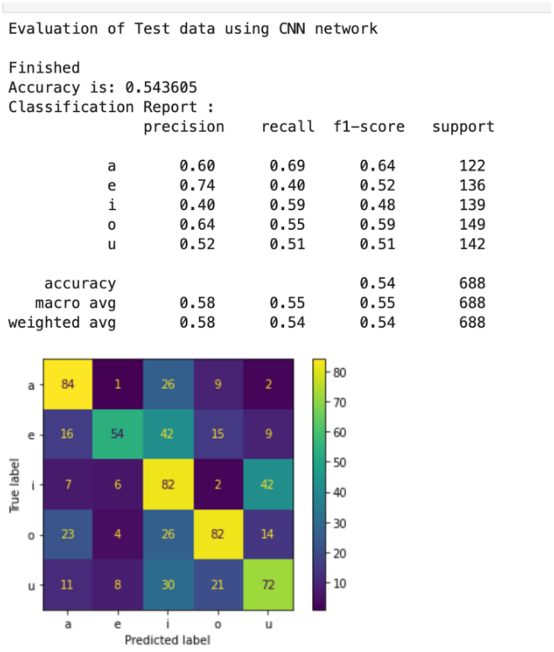
We attempt to classify the sound of vowel spoken. Here, we have 5 multiple classes: ‘a’, ‘e’, ‘I’, ‘o’, ‘u’. Since this is a multi-class classification of 1 of the 5 possible vowels, we use Cross Entropy Loss function and SoftMax function on the last layer output of both Networks and the output will be a vector of size 5 which contains probabilities for each of the 5 classes. Finally, we use the argmax function to pick the class index which corresponds to the highest probability to obtain the predicted class output.





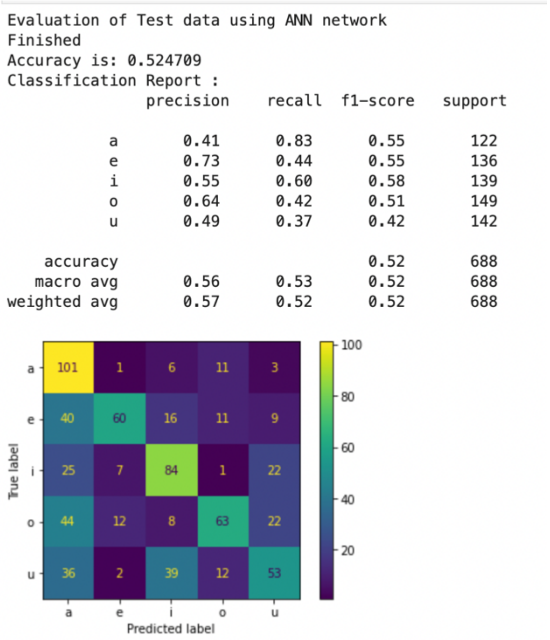
Above is the training and validation accuracy of both models as we trained the networks across epochs. The x-axis denotes epochs while y-axis denotes accuracy. We observe that training of the second Network B (shown on the right-hand-side) may require more epochs as we observe the higher accuracy values towards later epoch values.

Below is the classification report & confusion matrix plot for both models on the test data:



```
Evaluation of Test data using CNN network

Finished
Accuracy is: 0.543605
Classification Report :
               precision    recall  f1-score   support

           a       0.60      0.69      0.64       122
           e       0.74      0.40      0.52       136
           i       0.40      0.59      0.48       139
           o       0.64      0.55      0.59       149
           u       0.52      0.51      0.51       142

    accuracy                           0.54       688
   macro avg       0.58      0.55      0.55       688
weighted avg       0.58      0.54      0.54       688
```

True label \ Predicted label	a	e	i	o	u
a	84	1	26	9	2
e	16	54	42	15	9
i	7	6	82	2	42
o	23	4	26	82	14
u	11	8	30	21	72



```
Evaluation of Test data using ANN network
Finished
Accuracy is: 0.524709
Classification Report :
               precision    recall  f1-score   support

           a       0.41      0.83      0.55       122
           e       0.73      0.44      0.55       136
           i       0.55      0.60      0.58       139
           o       0.64      0.42      0.51       149
           u       0.49      0.37      0.42       142

    accuracy                           0.52       688
   macro avg       0.56      0.53      0.52       688
weighted avg       0.57      0.52      0.52       688
```

True label \ Predicted label	a	e	i	o	u
a	101	1	6	11	3
e	40	60	16	11	9
i	25	7	84	1	22
o	44	12	8	63	22
u	36	2	39	12	53

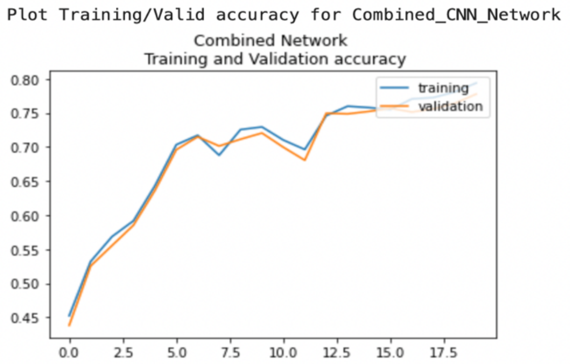
The accuracy comparison between the 2 models was relatively the same although Network A had a slightly higher accuracy score of 0.54 compared to Network B with 0.52. Network A was good at predicting vowel ‘A’ with an F1 score of 64%. Network B outperformed Network A in achieving an F1 score of 0.58 vs 0.48 on the vowel ‘I’. This result is promising as it shows us that a combination of these 2 models may complement each other for specific vowels and may help improve the overall accuracy.

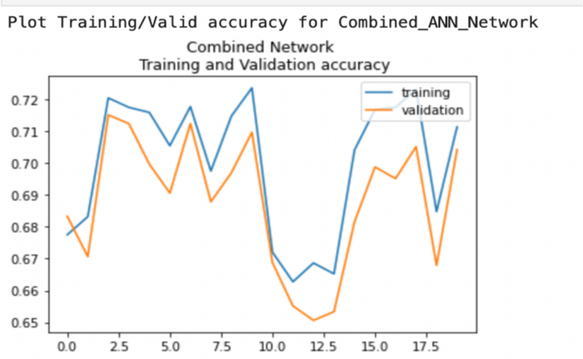
Combined models

I decided to combine both models into by creating 2 forward pass networks for both tasks where the output of the model is 2 outputs: the first is the gender and the second is the vowel classification. This required modification to have 2 labels saved as a list associated with each Mel-spectrogram e.g.: [gender class, vowel class]. Training the network, made use of 2 separate loss functions both BCE Loss and Cross-Entropy Loss. During the training loop of the backpropagations, the loss values were added together to produce a total loss to be used to update our model similar to how GAN models are trained. This forces our model to minimize the total loss and to produce a model that's able to perform well for both tasks.

An example output of the model with the two output labels is shown in appendix {3} where the 1st index is the prediction and the second is the target value: gender (predicted =1, actual= 1), and vowel (predicted =2, actual=4)

An alternative approach would have been to make use of the pre-trained models of Network A and Network B trained initially with and passed the appropriate datasets through them before combining/concatenating the results.





Comparison of training between the two Combined models show that the CNN Network produces a much steadier training curve as we observe the training accuracy increasing steadily across epochs. Note: the accuracies considered here are calculated for each of the classification task before being added together and divided by 2 to produce a mean accuracy for our model for both tasks.

Analysis of the results of the output of the Combined CNN model shows that we have been able to improve the accuracy of the Vowel classification aspect of our model from 0.54 to 0.58 when compared to the previous output of Network A. This is shown in the appendix {1} and {2}.

Conclusion and future works

From the 2 tasks explored above, we observe that the best model for each class was:

- Task 1: gender classification – Network A – CNN network
- Task 2: Classification of Vowel – Network A – CNN network
- Task 3: Combined Classification of gender and vowel classification – Combined Network A

The difference between the two explored networks for individual classification tasks was very little. The main advantage of the CNN networks was the use of Convolutional layers which provided an advantage over our ANN network. These Conv2D layers were able to extract better features from the mel-spectogram image which resulted in higher order features when the outputs were passed through a Linear layer for classification task.

We observe that using a Convolutional Layer is not always strictly necessary to obtain good performance on models since the difference between the two models on both classification tasks varied by only about 2-3%. Further exploration could be done to study the effects of increasing the number of linear layers in the CNN model.

Future work could explore using the larger dataset available that contains over 10 hours of audio [2] experimentation in optimal batch size to improve prediction accuracy, speed and finally training for more epochs give more computational resources. It may be possible to use smaller audio clip lengths for gender classification since pitch and tone is normally differentiator between male and female voices therefore, we can do data augmentation to produce more training data. For vowel, it may require using the full recording since some vowels depending on the singing technique sound very similar.

Our models could also be compared to a model using a pre-trained model such as VVG or ResNet model to compare performance to see whether using a pre-trained can help with our small training data limitation and give a boost in performance for the CNN models. Further experimentation and optimization on initial parameter values used to construct the input Mel-spectrograms images could be explored which would help find the best Mel-spectrogram input representations that would produce the best improvement in accuracy of models. One could also explore combinations of both Networks together e.g., Network A and Network B in one combined model to see the effect of the outputs.

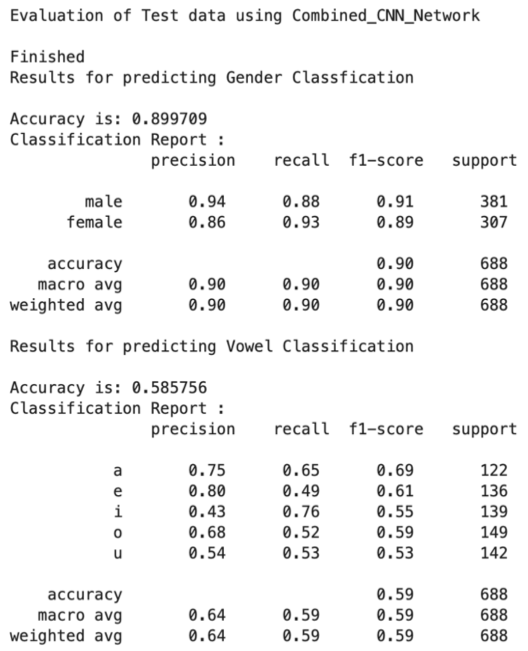
References

[1] - Vocal Dataset – (Found on JHub shared access – shared storage/EC7013P/vocal_by_vowel directory)

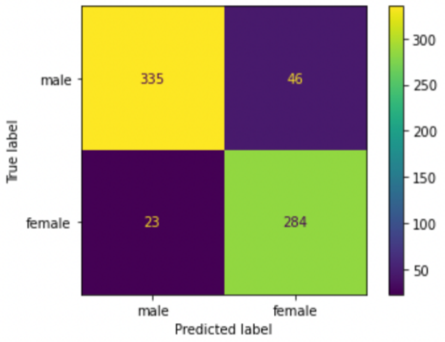
[2] - Wilkins, Julia, Prem Seetharaman, Alison Wahl, & Bryan Pardo. (2018). VocalSet: A Singing Voice Dataset (1.2) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.1442513

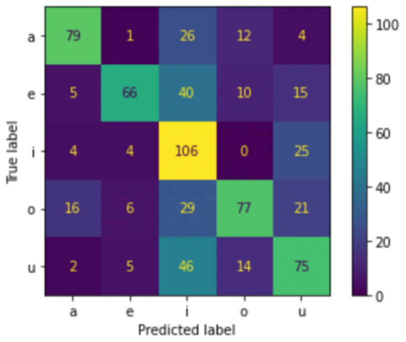
Appendix:

1. Combined CNN Network Output

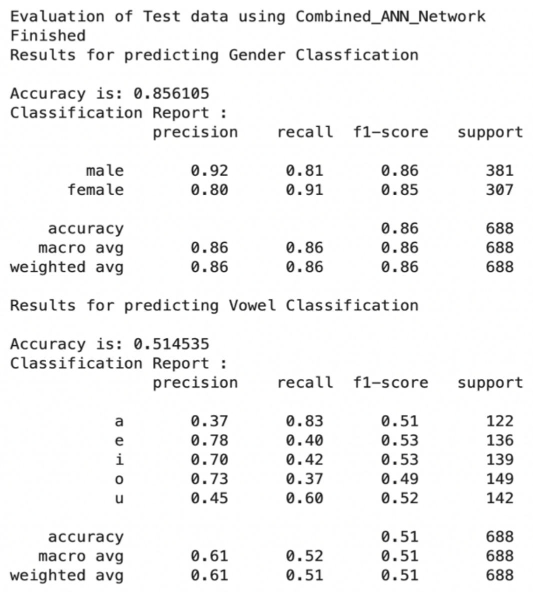


```
Evaluation of Test data using Combined_CNN_Network

Finished
Results for predicting Gender Classfication

Accuracy is: 0.899709
Classification Report :
               precision    recall  f1-score   support

        male       0.94      0.88      0.91       381
      female       0.86      0.93      0.89       307

    accuracy                           0.90       688
   macro avg       0.90      0.90      0.90       688
weighted avg       0.90      0.90      0.90       688

Results for predicting Vowel Classification

Accuracy is: 0.585756
Classification Report :
               precision    recall  f1-score   support

           a       0.75      0.65      0.69       122
           e       0.80      0.49      0.61       136
           i       0.43      0.76      0.55       139
           o       0.68      0.52      0.59       149
           u       0.54      0.53      0.53       142

    accuracy                           0.59       688
   macro avg       0.64      0.59      0.59       688
weighted avg       0.64      0.59      0.59       688
```

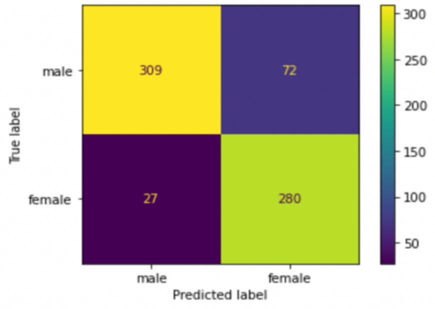


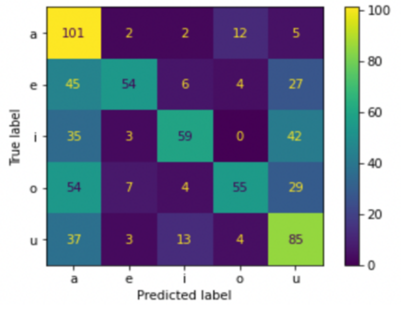


2. Combined ANN Network Output

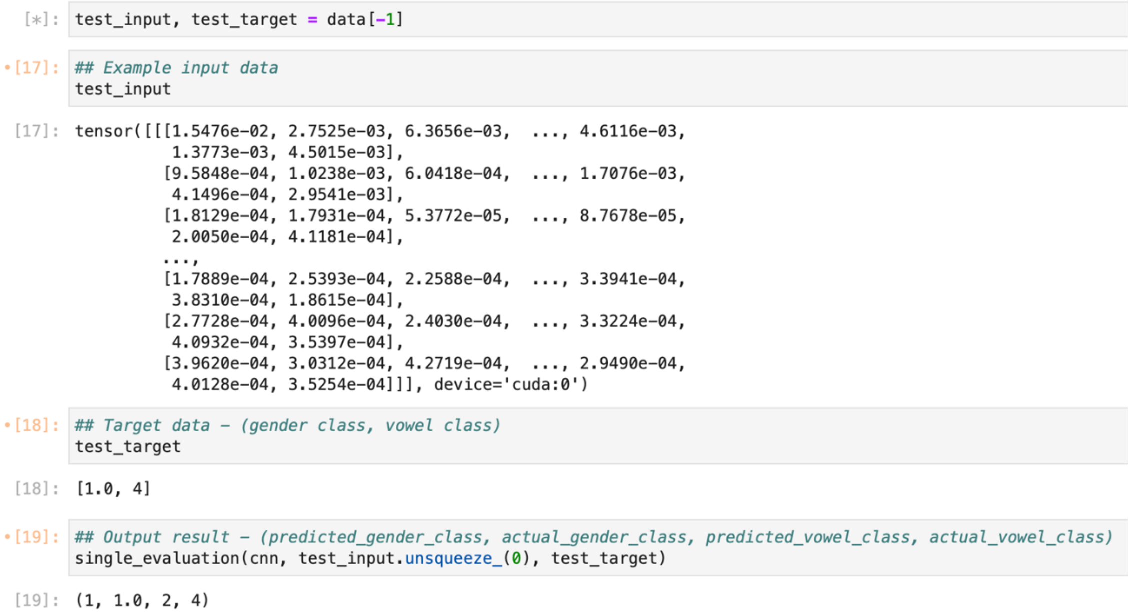


```
Evaluation of Test data using Combined_ANN_Network
Finished
Results for predicting Gender Classfication

Accuracy is: 0.856105
Classification Report :
               precision    recall  f1-score   support

        male       0.92      0.81      0.86       381
      female       0.80      0.91      0.85       307

    accuracy                           0.86       688
   macro avg       0.86      0.86      0.86       688
weighted avg       0.86      0.86      0.86       688

Results for predicting Vowel Classification

Accuracy is: 0.514535
Classification Report :
               precision    recall  f1-score   support

           a       0.37      0.83      0.51       122
           e       0.78      0.40      0.53       136
           i       0.70      0.42      0.53       139
           o       0.73      0.37      0.49       149
           u       0.45      0.60      0.52       142

    accuracy                           0.51       688
   macro avg       0.61      0.52      0.51       688
weighted avg       0.61      0.51      0.51       688
```





3. Example output of single evaluation of data example using Combined_CNN_Network



```
[*]: test_input, test_target = data[-1]

•[17]: ## Example input data
test_input

[17]: tensor([[[1.5476e-02, 2.7525e-03, 6.3656e-03,  ..., 4.6116e-03,
           1.3773e-03, 4.5015e-03],
          [9.5848e-04, 1.0238e-03, 6.0418e-04,  ..., 1.7076e-03,
           4.1496e-04, 2.9541e-03],
          [1.8129e-04, 1.7931e-04, 5.3772e-05,  ..., 8.7678e-05,
           2.0050e-04, 4.1181e-04],
          ...,
          [1.7889e-04, 2.5393e-04, 2.2588e-04,  ..., 3.3941e-04,
           3.8310e-04, 1.8615e-04],
          [2.7728e-04, 4.0096e-04, 2.4030e-04,  ..., 3.3224e-04,
           4.0932e-04, 3.5397e-04],
          [3.9620e-04, 3.0312e-04, 4.2719e-04,  ..., 2.9490e-04,
           4.0128e-04, 3.5254e-04]]], device='cuda:0')

•[18]: ## Target data - (gender class, vowel class)
test_target

[18]: [1.0, 4]

•[19]: ## Output result - (predicted_gender_class, actual_gender_class, predicted_vowel_class, actual_vowel_class)
single_evaluation(cnn, test_input.unsqueeze_(0), test_target)

[19]: (1, 1.0, 2, 4)
```