B.TECH PROJECT

HANDWRITTEN SYLLABLE CLASSIFICATION-A NEGLECTED AREA OF REASEARCH

2020IMT-026

DEVANSHU PATIDAR

SUPERVISOR- PROF. ADITYA TRIVEDI

CONTENT

CHARACTER SET FOR A LANGUAGE

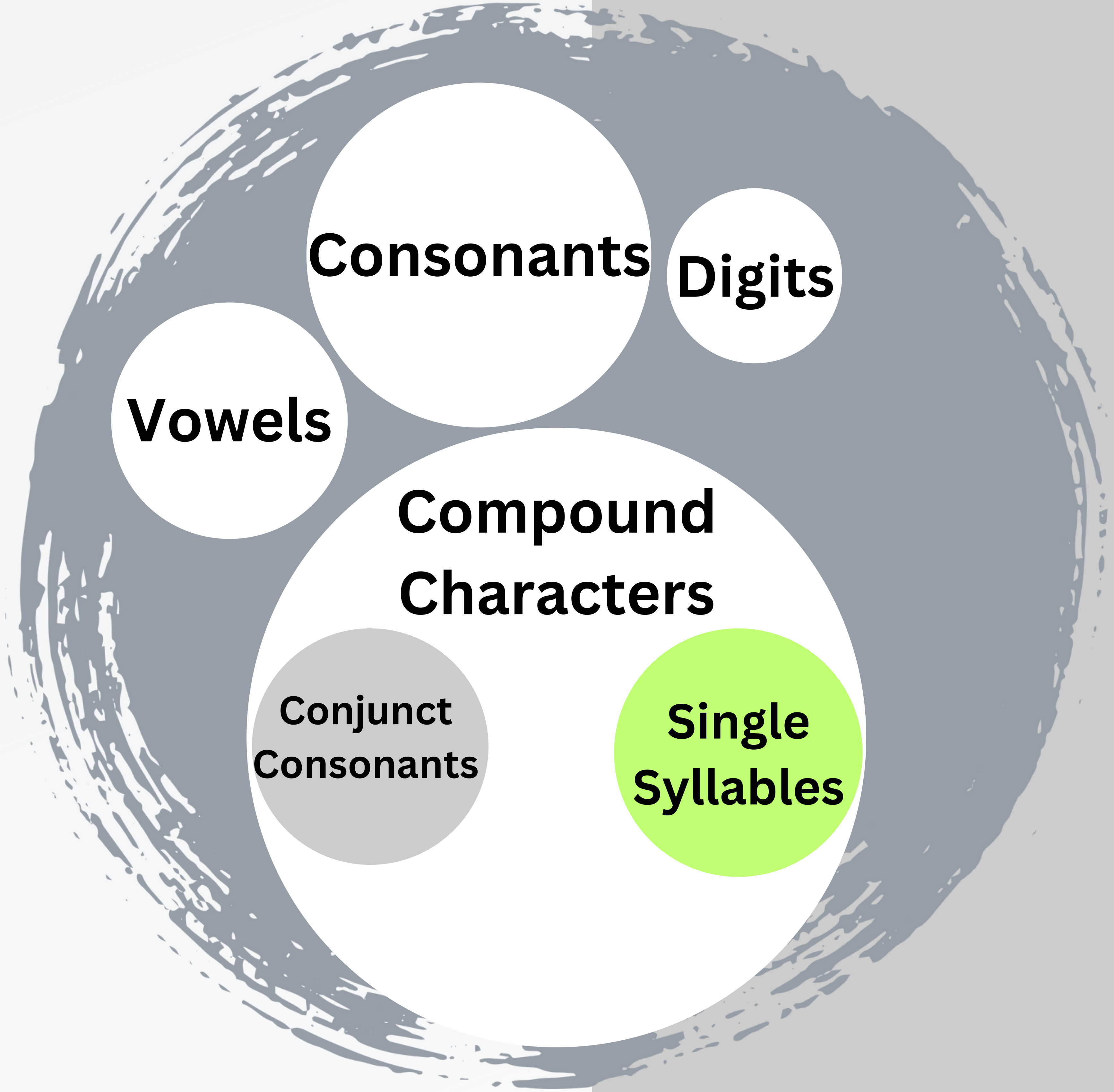

Figure 1: Character Set for most of the regional languages of India

COMPARISON

26 Alphabets and
10 digits
26+10 = 36 classes

ENGLISH

13 vowels, 33
consonants and 10
digits
13+ 13*33 + 33*33 +
10 = 1541 classes

GUJARATI

Figure 2: Comparison between English and Gujarati language

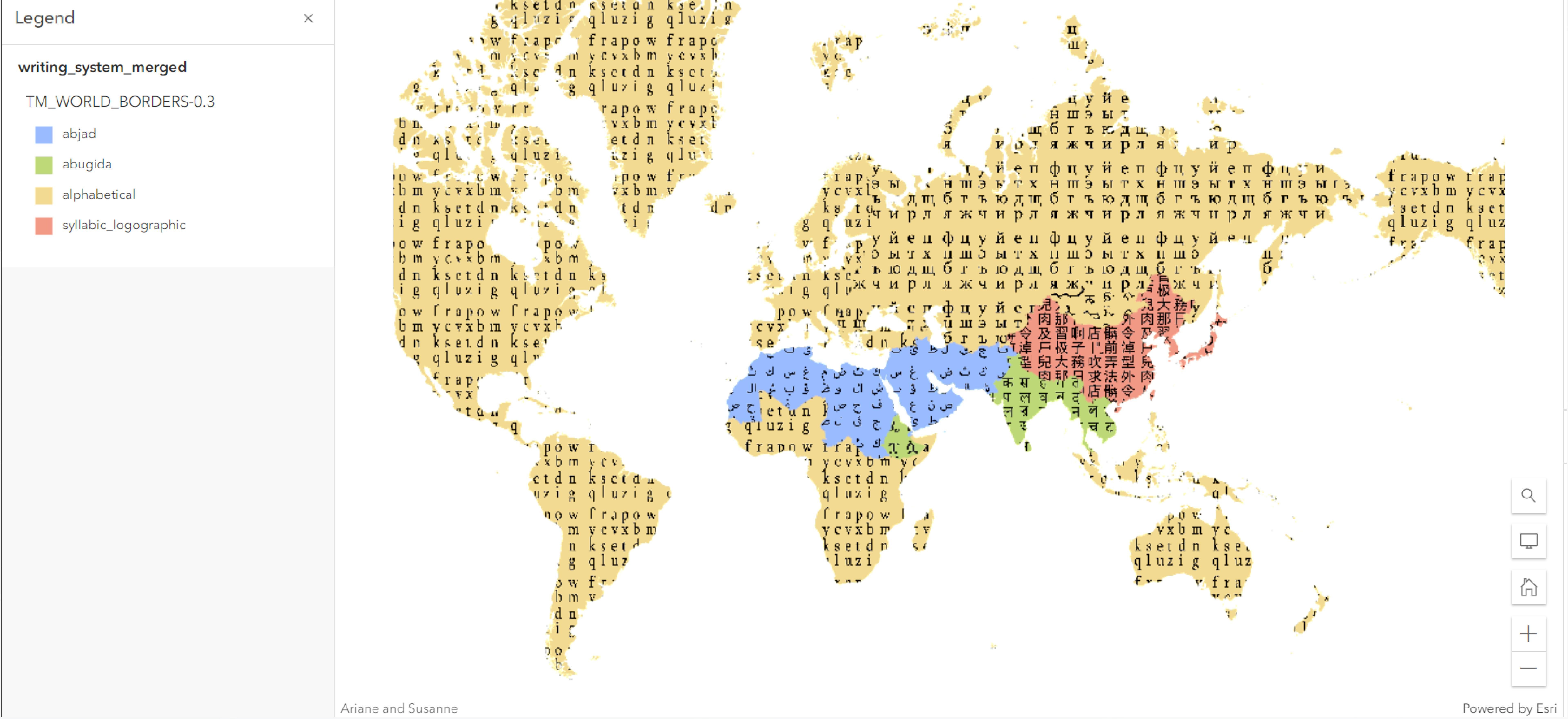

Figure 3: Major Writing systems in World[1]

SYLLABLES

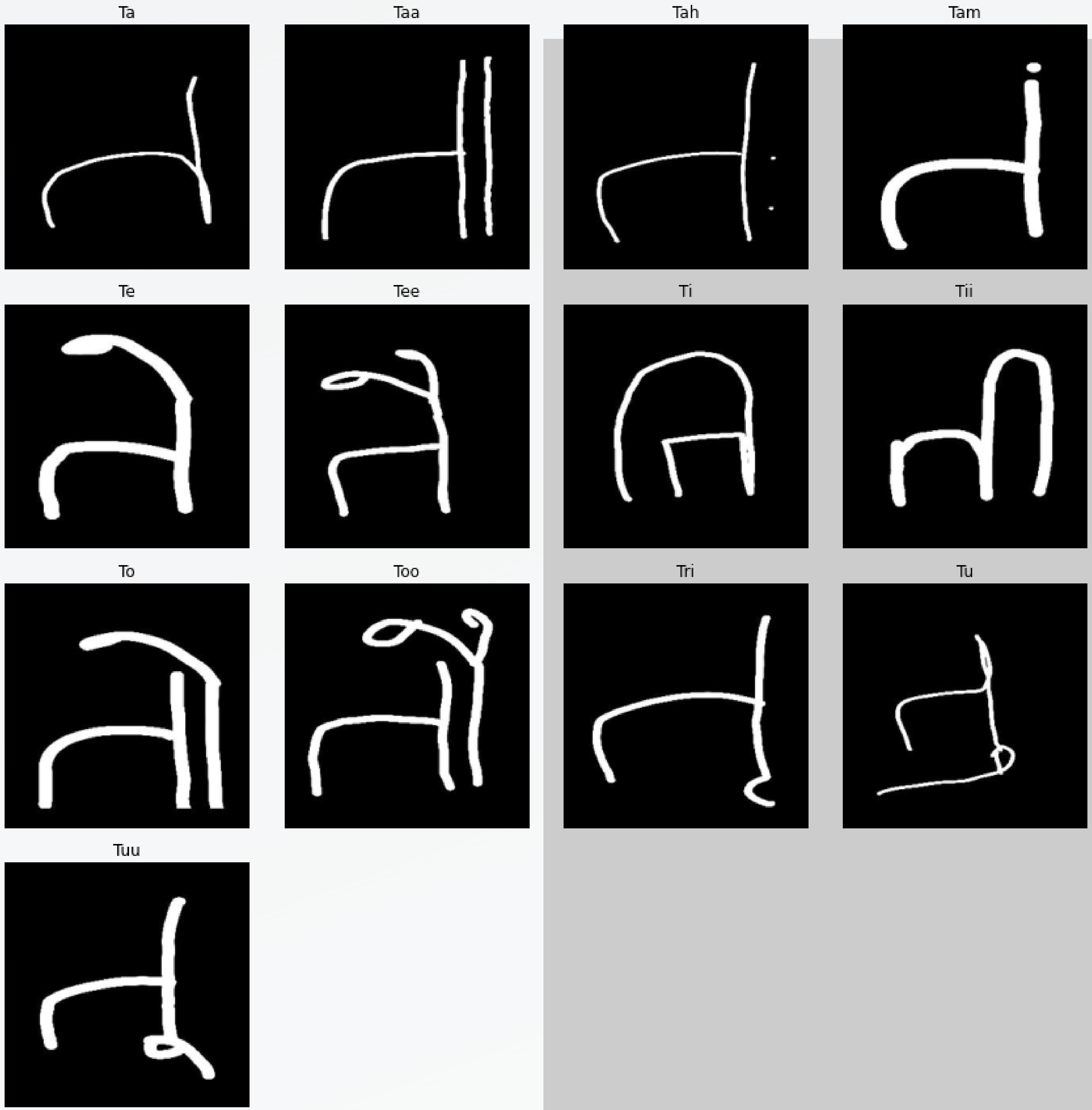

Figure 4: Different Classes in the Handwritten Syllable Dataset

MOTIVATION

- Document Digitalization- Bank Documents, Birth Certificates, Regional texts[2]
- What is not used, dies
- Will lead to advancements in Handwritten Character Recognition in Regional languages

RESEARCH GAPS

- LACK OF EXISTING PUBLICLY AVAILABLE DATASETS
- LACK OF EXISTING RESEARCH WORK ON HANDWRITTEN SYLLABLES
- USE OF ONLY TESTING DATASET FOR BOTH VALIDATION AND TESTING[2].
- LACK OF SMART CLASSIFICATION

OBJECTIVES

- TO GENERATE NEW DATASET
- TO PROVE THAT THE SYLLABLE CLASSES ARE VISUALLY SIMILAR
- TO PERFORM THE ACTUAL TEST

METHODOLOGY

DATASET GENERATION

- IMPORTED DRAWING BOARD
- ADDED SAVE AND CLEAR BUTTON
- SETS STROKE WIDTH RANDOMLY AND CLEARS THE CANVAS AUTOMATICALLY WHEN SAVE BUTTON IS PRESSED.

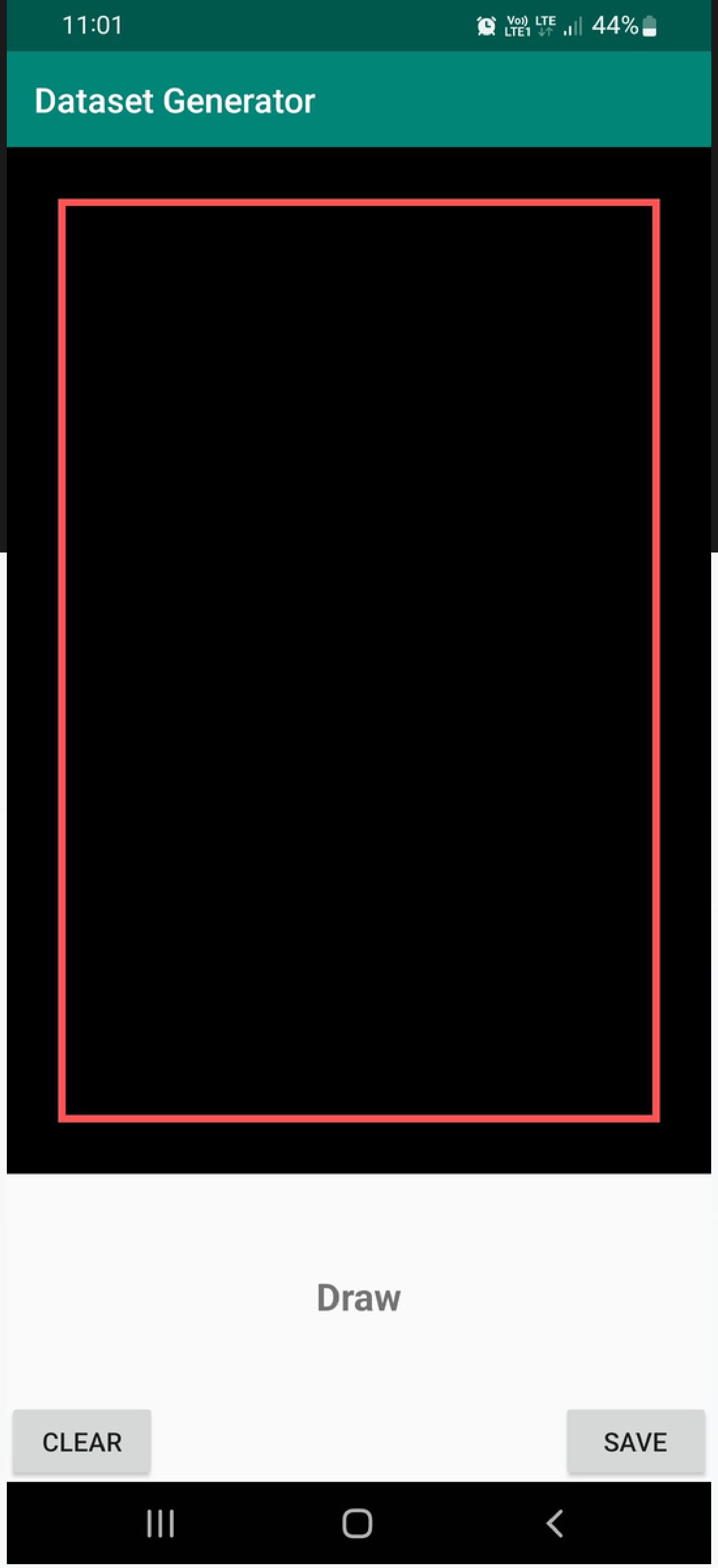

Figure 5: The app interface

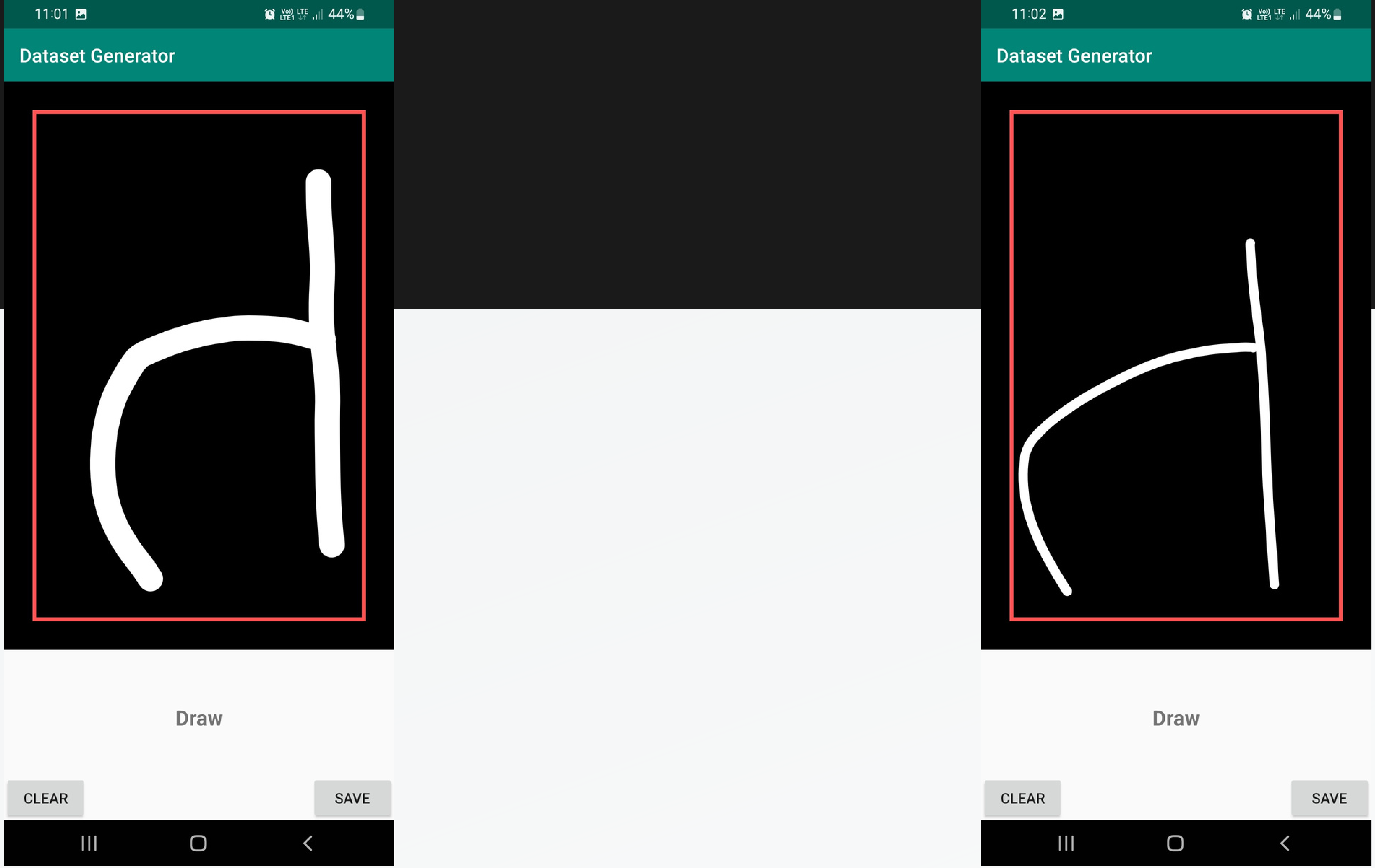

Figure 6: Image Displaying the random stroke width functionality

```kotlin
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)

    // Setup view instances.
    drawView = findViewById(R.id.draw_view)
    drawView?.setStrokeWidth(70.0f)
    drawView?.setColor(Color.WHITE)
    saveButton = findViewById(R.id.save_button)
    drawView?.setBackgroundColor(Color.BLACK)
    clearButton = findViewById(R.id.clear_button)
    drawTextView = findViewById(R.id.draw_text)
```

```kotlin
    // Setup clear drawing button.
    clearButton?.setOnClickListener { it: View!
        drawView?.clearCanvas()
        drawTextView?.text = getString(R.string.prediction_text_placeholder)
    }
```

Figure 7: Code Snippets related to the app

```kotlin
drawView?.setOnTouchListener { _, event ->
    // As we have interrupted DrawView's touch event,
    // we first need to pass touch events through to the instance for the drawing to show up.
    drawView?.onTouchEvent(event)
    true
}
```

```kotlin
// Setup save drawing button.
saveButton.setOnClickListener {
    val randomWidth = (24..81).random().toFloat()
    drawView?.setStrokeWidth(randomWidth)
    val initialBitmap = drawView?.getBitmap()
    if (initialBitmap != null) {
        val compressedBitmap = Bitmap.createScaledBitmap(initialBitmap, 256, 256, false)
        saveImage(compressedBitmap)
    } else Toast.makeText(this, "No Image Found", Toast.LENGTH_SHORT).show()
    drawView?.clearCanvas()
}
```

Figure 8: Code snippets related to the app part 2

DATASET

- 84 images for Train Set.
- 18 for Validation Set.
- 18 for Testing Set.

IMAGE PADDING

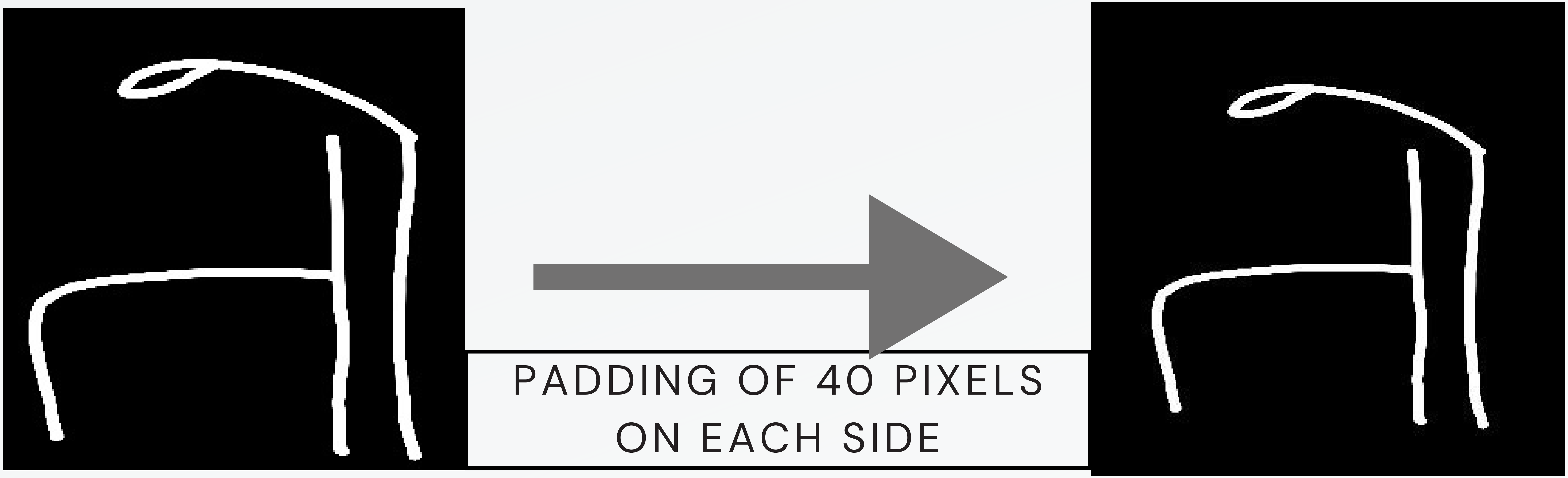

Figure 9: Figure showing image before and after padding

DATA PREPROCESSING

- IMAGE RESIZING
- MAKING THE IMAGE RGB FROM GREYSCALE
- LABELING IMAGES BASED ON CLASS DIRECTORIES
- LABELING IS CONVERTED TO ONE HOT LABELING

DATA AUGMENTATION

```
rotation_range=15,
width_shift_range=0.10,
height_shift_range=0.10,
shear_range=0.25,
zoom_range=0.1,
rescale=1.0 / 255.0
```

ONLY ON TRAINING SET ,NOT ON TESTING OR VALIDATION SET

SMALL DATASET -HOW TO AVOID OVERFITTING

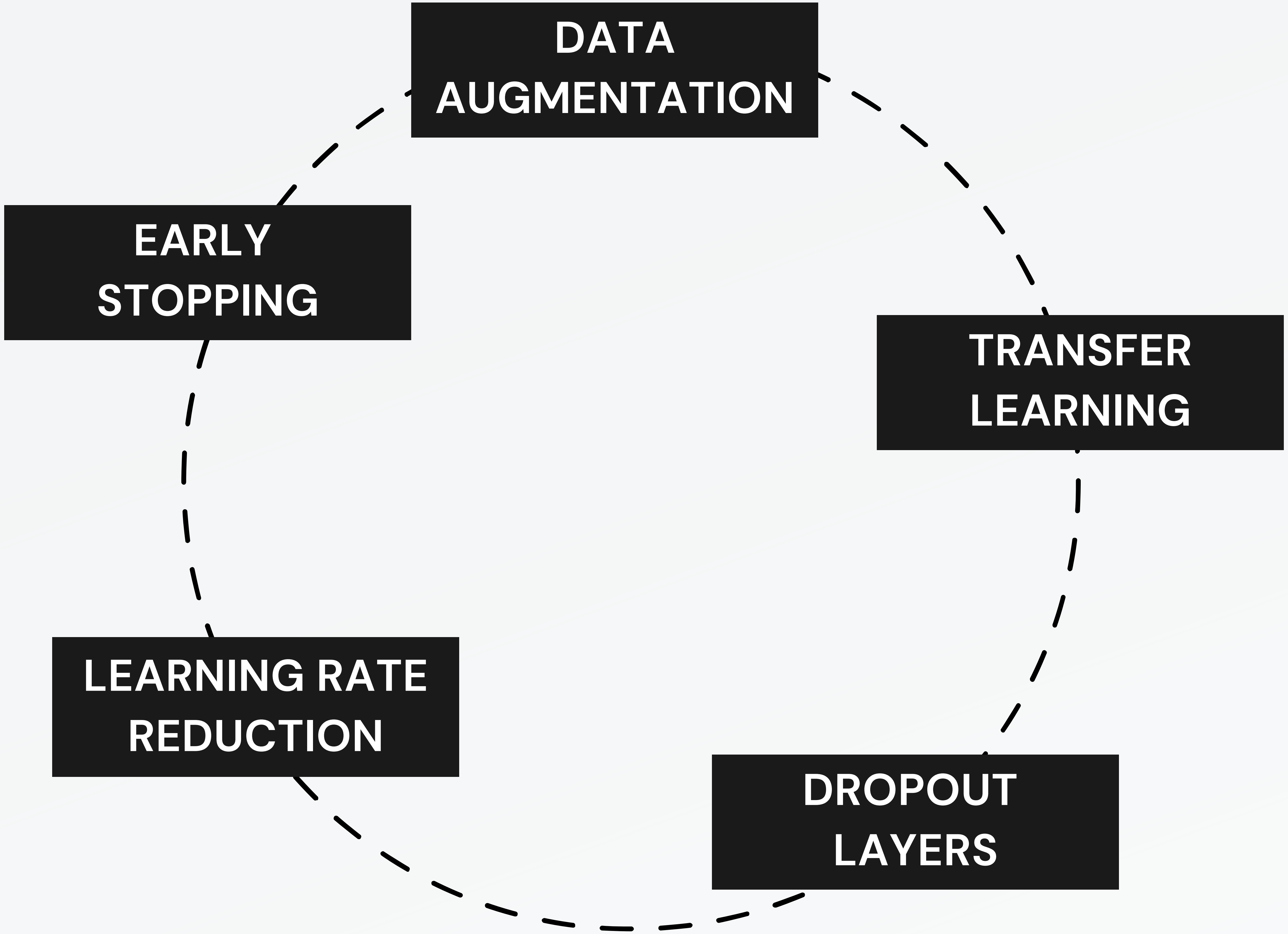

Figure 10: Techniques employed to overcome overfitting

MODELS CHOSEN FOR COMPARISION

- EfficientNetv2s model is designed to achieve high performance with relatively fewer parameters, making it efficient in terms of memory and computation.
- InceptionV3 is a deep convolutional neural network architecture that was developed by Google's research team. It's a successor to the original Inception (GoogLeNet) architecture and is designed specifically for tasks like image classification.
- Xception model is particularly wellsuited for image-related tasks like image classification, object detection, and image segmentation.

PROPOSED ARCHITECTURE

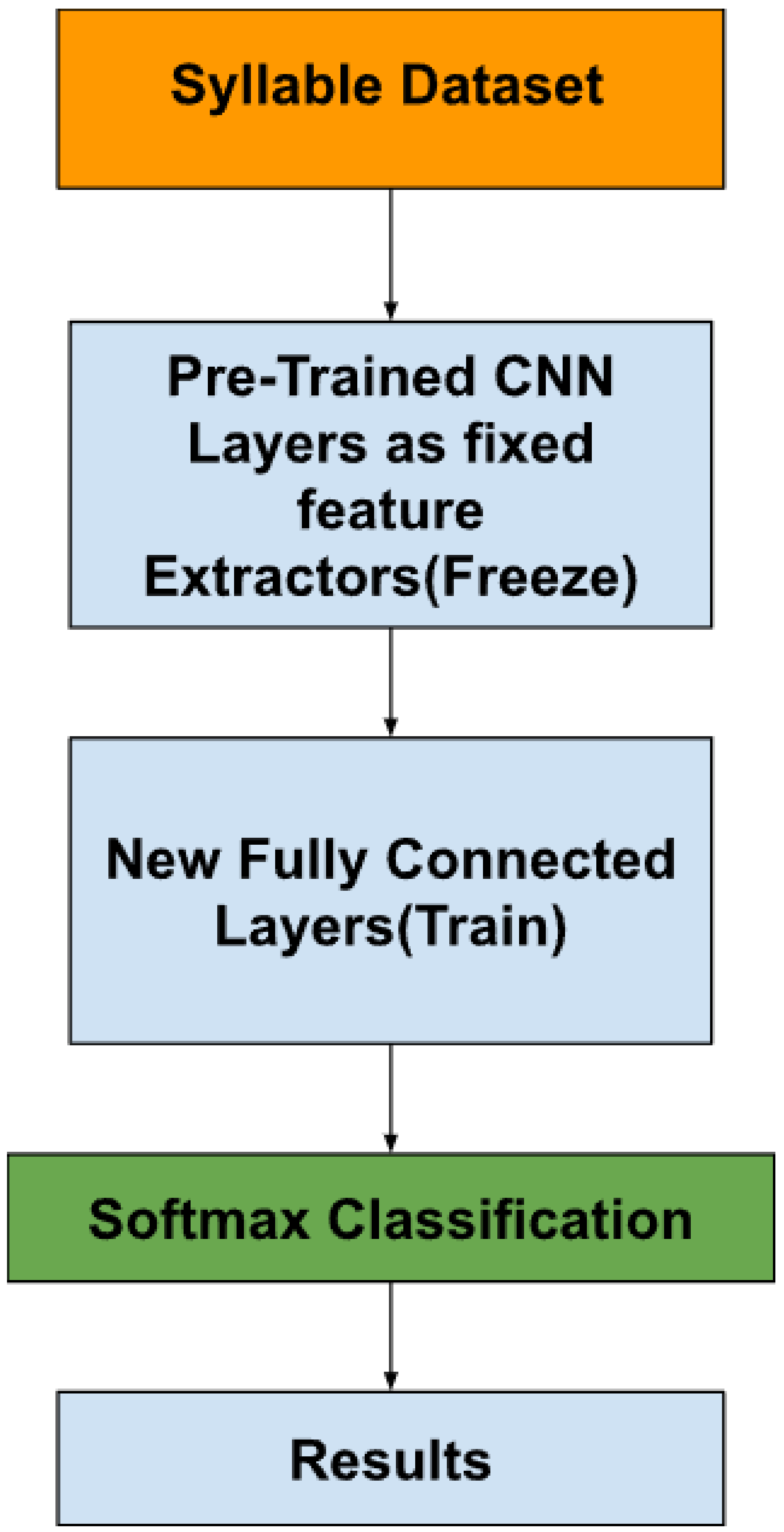

Figure 11: The proposed Architecture[2]

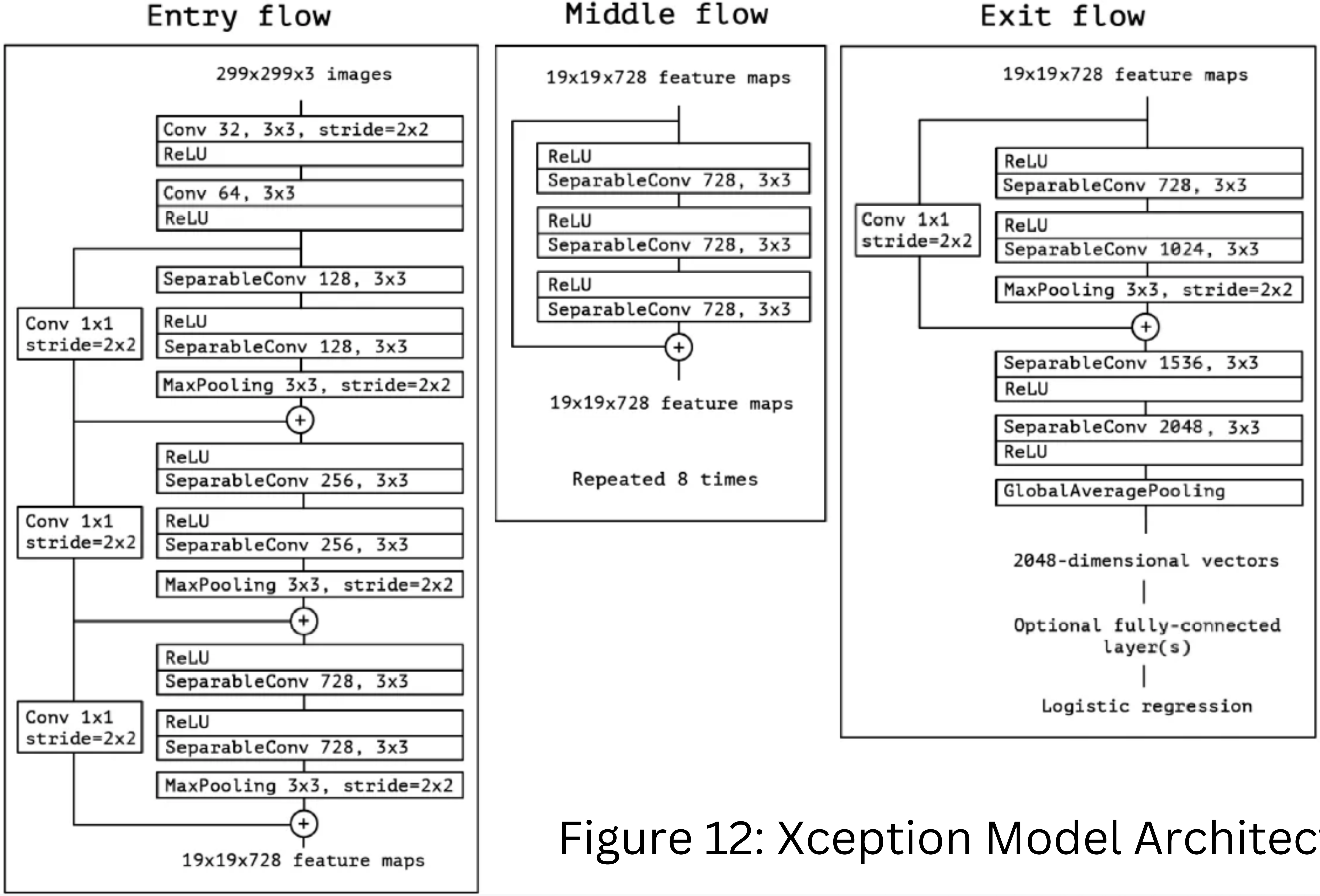

Figure 12: Xception Model Architecture[3]

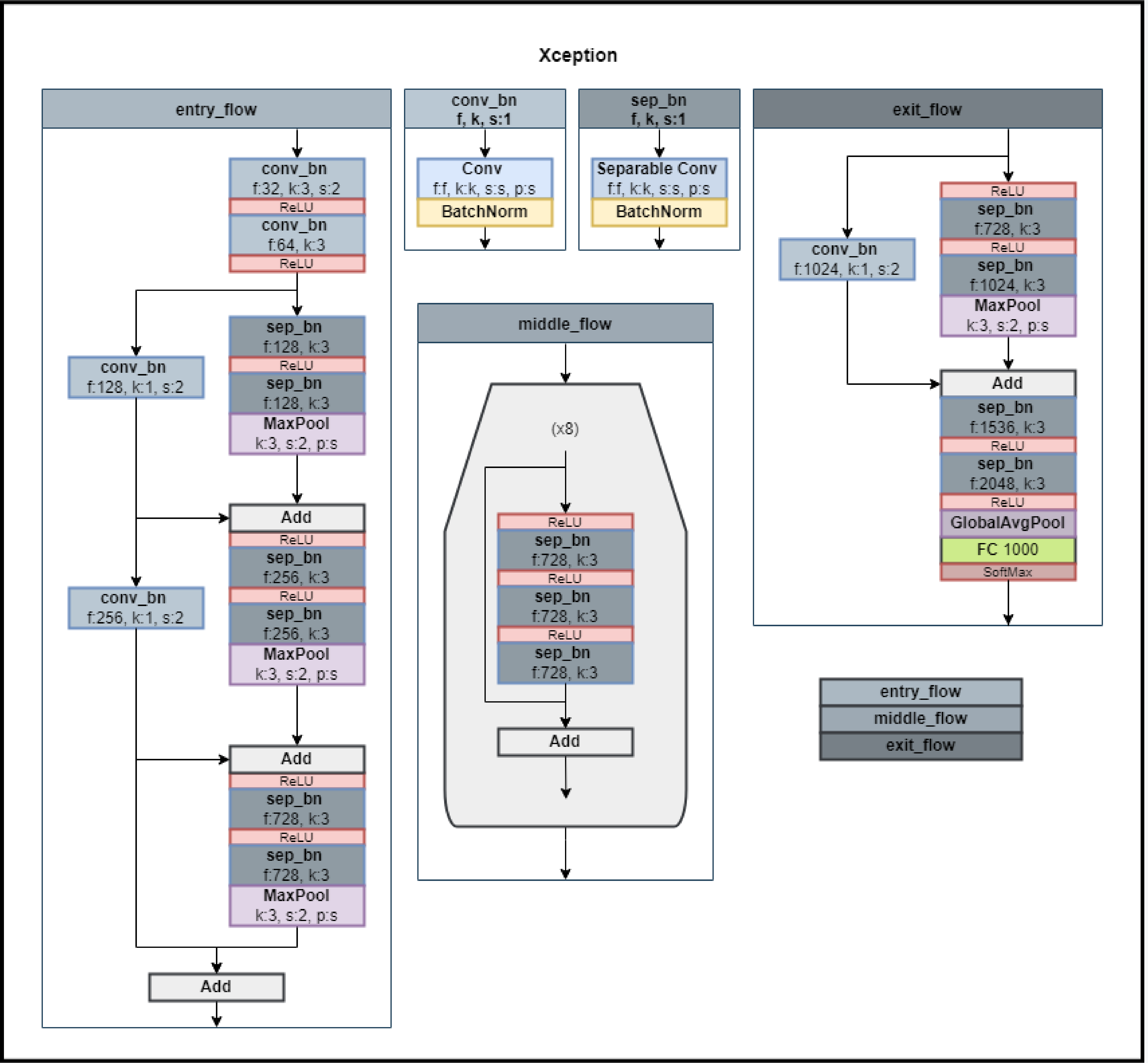

Figure 13: Xception Model Architecture[3]

MODEL SUMMARY

```
________________________________________________________________
 Layer (type)                Output Shape              Param #
================================================================
 xception (Functional)       (None, 4, 4, 2048)        20861480

 global_average_pooling2d_1  (None, 2048)              0
  (GlobalAveragePooling2D)

 dense_3 (Dense)             (None, 512)               1049088

 dropout_2 (Dropout)         (None, 512)               0

 dense_4 (Dense)             (None, 256)               131328

 dropout_3 (Dropout)         (None, 256)               0

 dense_5 (Dense)             (None, 13)                3341

================================================================
Total params: 22045237 (84.10 MB)
Trainable params: 1183757 (4.52 MB)
Non-trainable params: 20861480 (79.58 MB)
```

Figure 14: Model summary for Xception

ACTIVATION FUNCTION USED

SOFTMAX ACTIVATION FUNCTION

It is used to convert the raw scores produced by the previous layer into a probability distribution over multiple classes.

$$softmax(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}}$$

Figure 15: Maths behind Softmax Activation Function[4]

LOSS FUNCTION USED

CATEGORICAL CROSSENTROPY

It measures the dissimilarity between the predicted probability distribution and the true distribution of the target classes.
Loss over all training examples in computed and averaged.

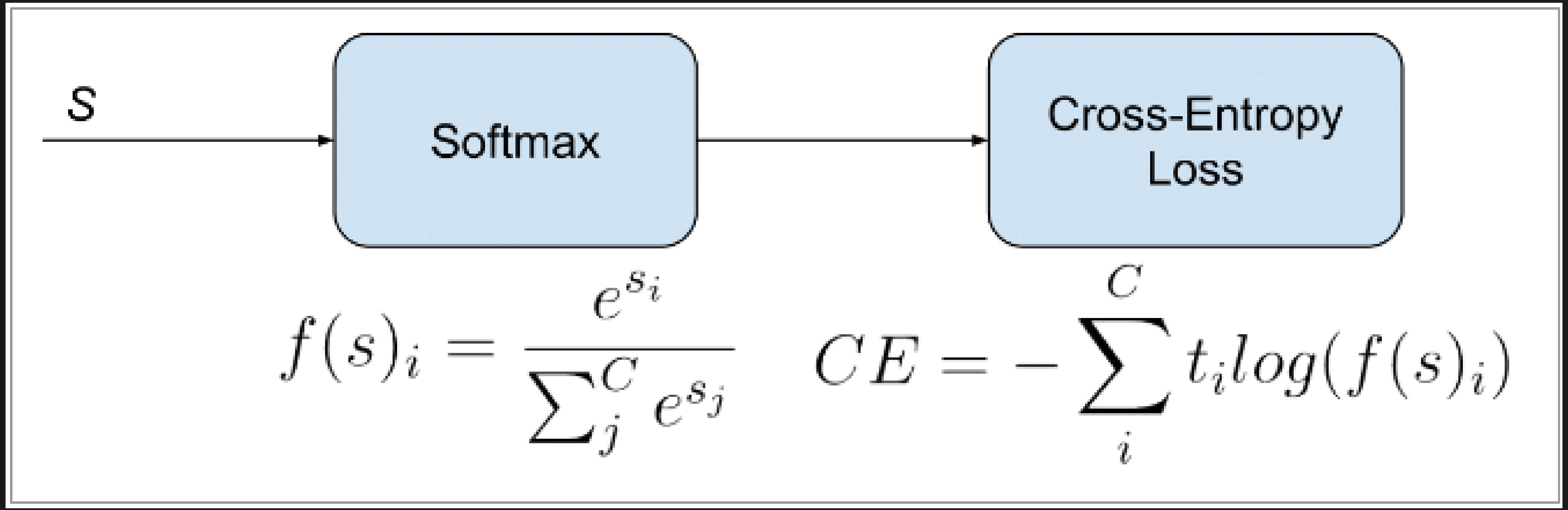

Figure 16: Maths behind Categorical Crossentropy[5]

T-SNE PLOTS

- Data points of the same class are not so well clustered closer to each other than the other classes. In fact ,separation is not clear.

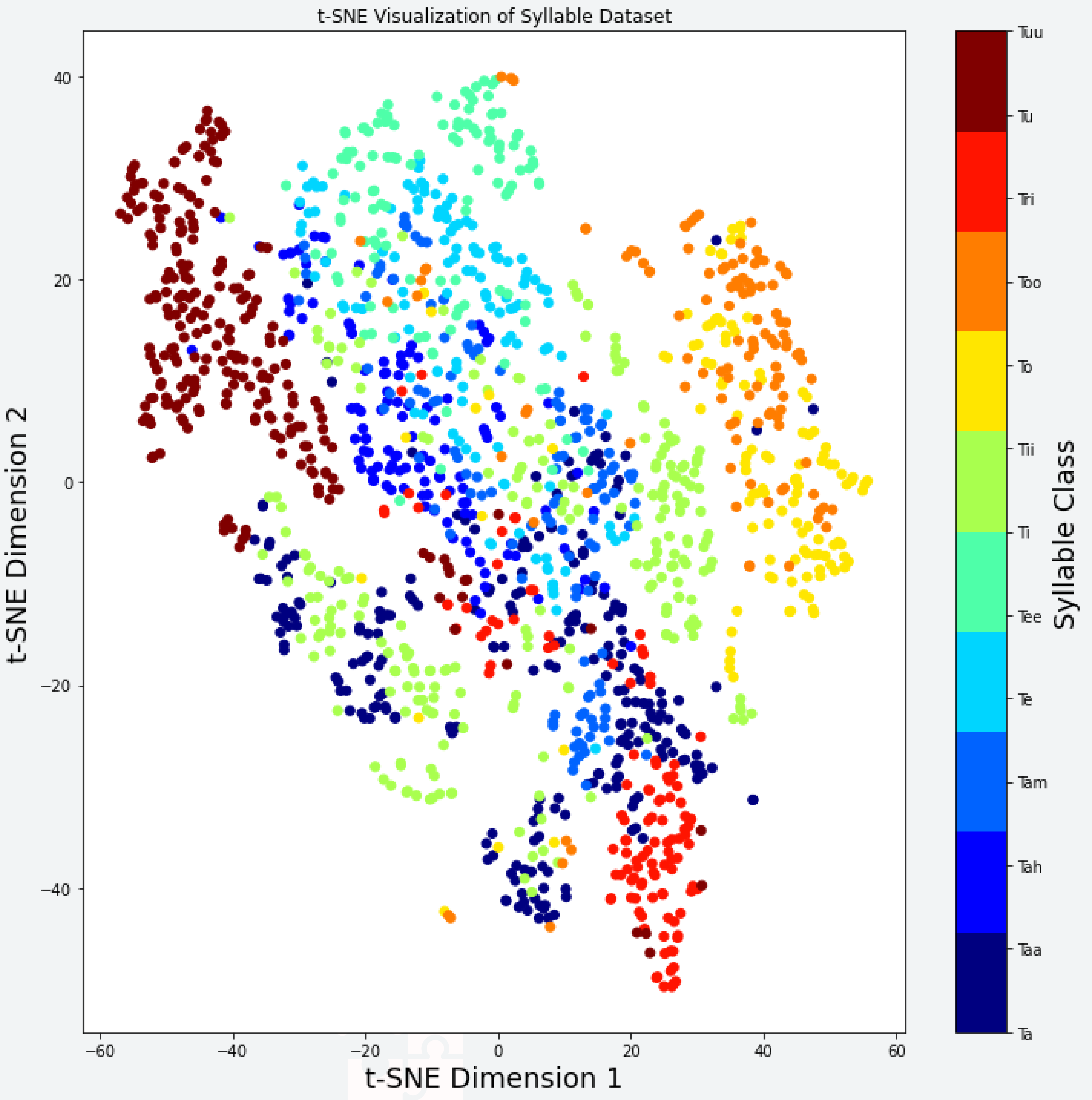

Figure 17: t-SNE plot for Syllable Classes

- Data points of the same class are clustered closer to each other than the other classes.

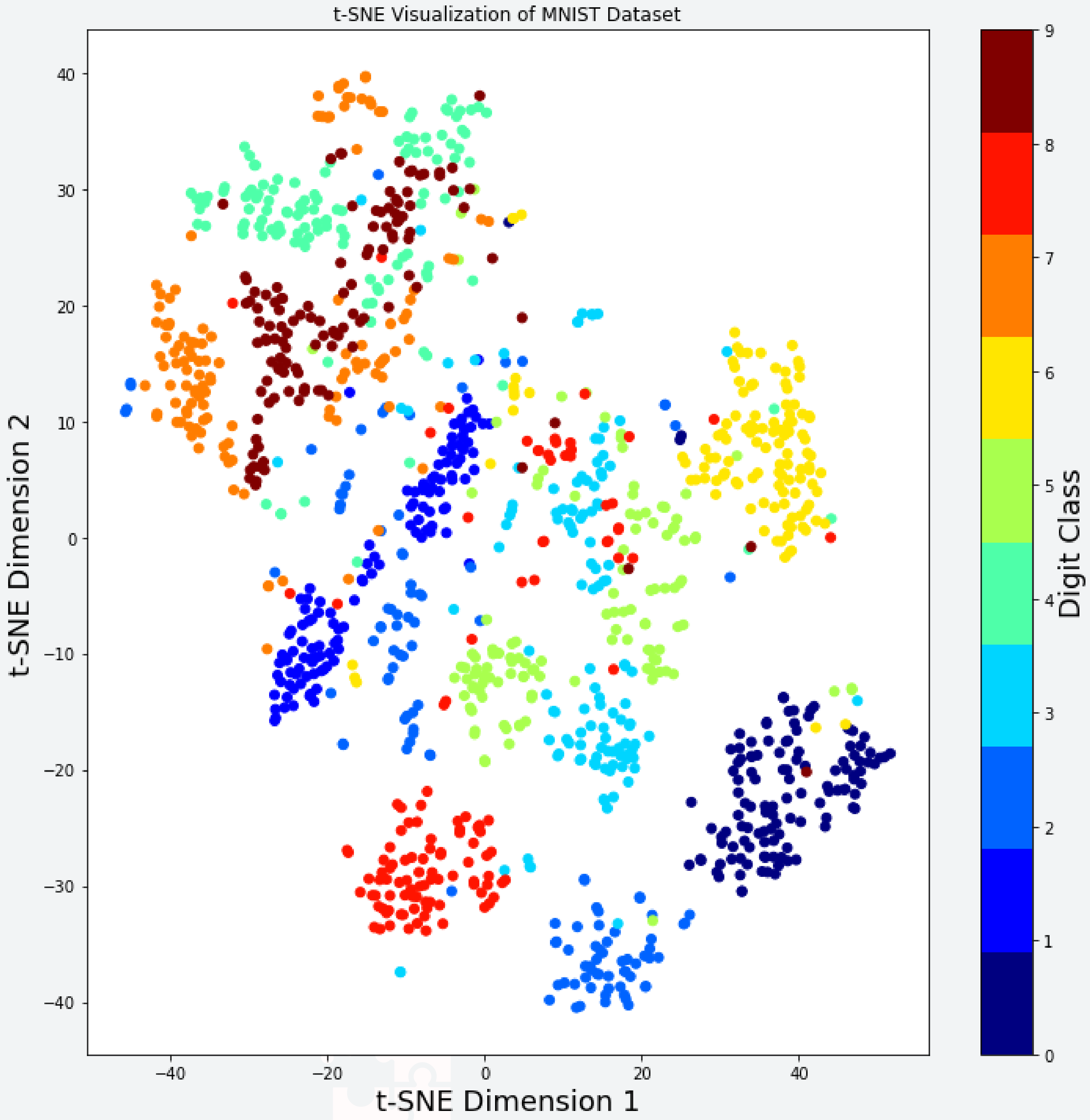

Figure 18: t-SNE plot for Mnist Digit Classes

UMAP PLOTS

- Data points of the same class are not so well clustered closer to each other than the other classes. In fact ,separation is not clear.

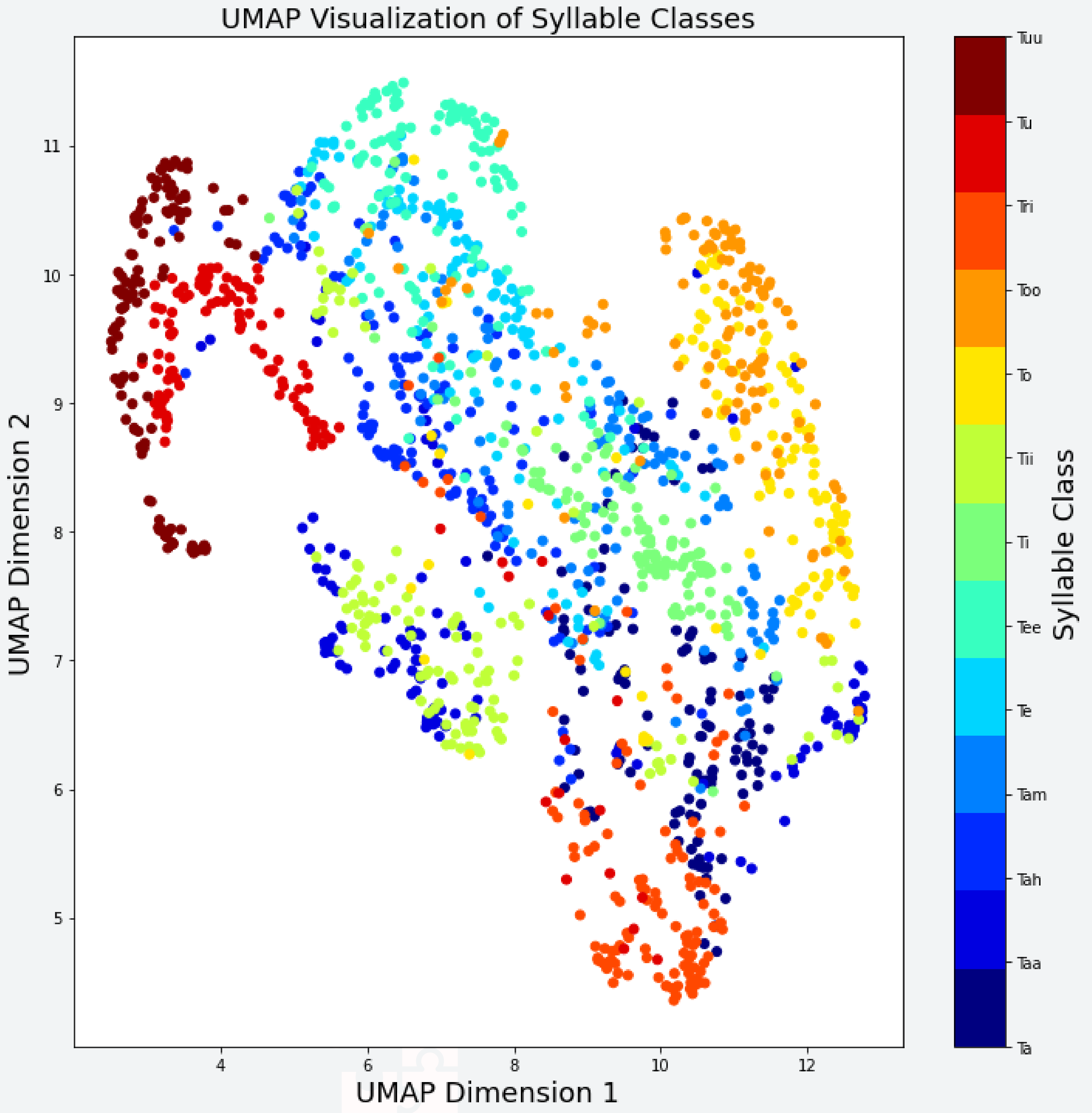

Figure 19: UMAP plot for Syllable Classes

- Data points of the same class are clustered closer to each other than the other classes.

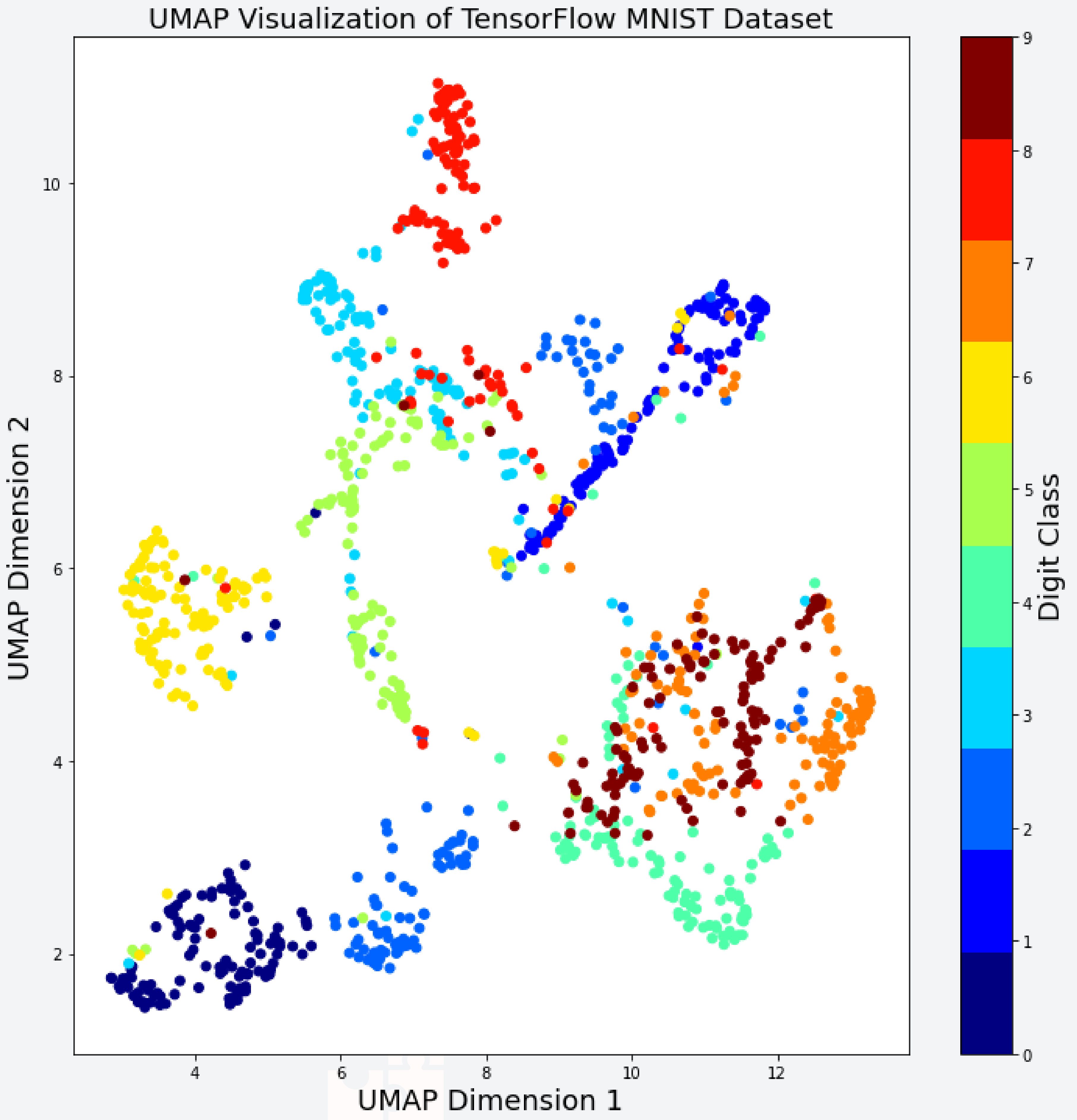

Figure 20: UMAP plot for Mnist Digit Classes

RESULTS

METRICS USED

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

$$\text{Precision (P)} = \frac{TP}{TP + FP}$$

$$\text{Recall (R)} = \frac{TP}{TP + FN}$$

$$\text{AUC} = \frac{TP/(TP + FN) + TN/(TN + FP)}{2}$$

Figure 21: Performance Metrics used[6]

Table 1: Training Results

Model Used	Train Acc (%)	Test Acc(%)	Precision(%)	Recall(%)	AUC(%)
Xception	96.52	92.65	92.57	92.3	99.71
InceptionV3	94.59	91.02	92.65	91.02	99.76
EfficientNetV2S	90.29	85.47	88.72	87.47	99.30

A combination of high accuracy, precision, recall, and AUC scores collectively indicates that the model has not only achieved accurate overall predictions but has also developed a strong understanding of data patterns, enabling it to make reliable predictions on new data points while distinguishing between different classes effectively.

CONFUSION MATRIX

- HIGH MAIN DIAGONAL VALUES IN THE CONFUSION MATRIX
- IT SUGGESTS THE MODEL IS CORRECTLY CLASSIFYING INSTANCES, MAINTAINING A LOW RATE OF FALSE POSITIVES AND FALSE NEGATIVES.

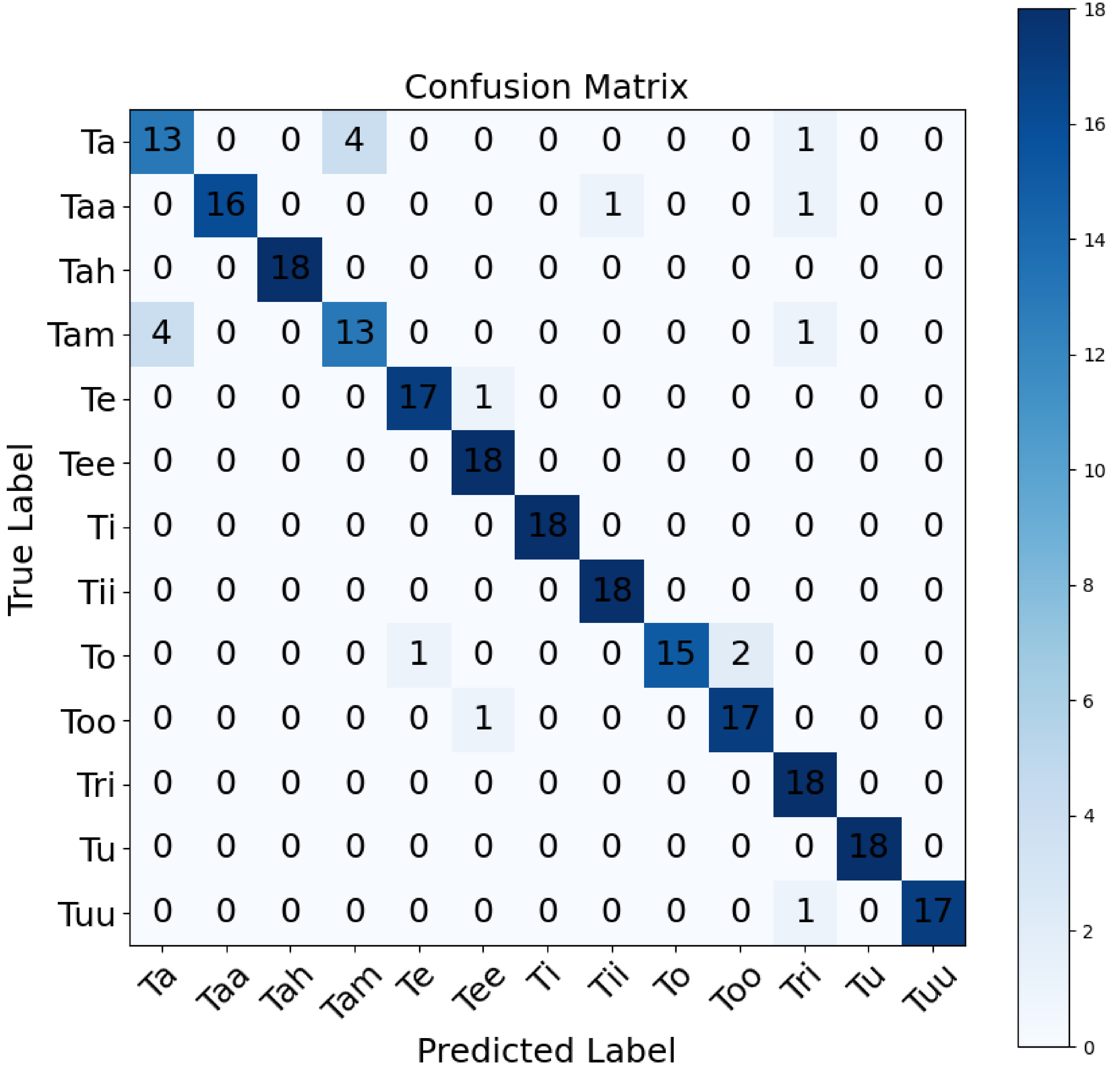

Figure 22: Confusion Matrix for Xception Model

IMPORTANT NOTE

- It is essential to realize the model's performance may not necessarily generalize well to larger, more diverse datasets. As such, further validation on larger datasets is needed.

FUTURE SCOPE

- DATASET IMPROVEMENTS- ADD MORE VARIABILITY, USE OF SYNETHIC IMAGES
- TRAINING FROM SCRATCH OR TESTING ON MORE TRANSFER LEARNING SCENARIOS.
- COMPARISON WITH OTHER POPULAR MODELS
- ADDITION OF A THRESHOLD TO CLASSIFY FALSE INPUTS AS ERROR.

REFERENCES

[1] A. Geograz, “Image showing writing systems and scripts of the world.” https://www.arcgis.com/home/item.html?id=c12011864d984eabbac8d9d0136d1066. Last Accessed on: Aug. 21, 2023

[2]P. Goel and A. Ganatra, "Handwritten Gujarati Numerals Classification Based on Deep Convolution Neural Networks Using Transfer Learning Scenarios," in IEEE Access, vol. 11, pp. 20202-20215, 2023, doi: 10.1109/ACCESS.2023.3249787.

[3]“CNN-Architectures/Implementations/Xception at master · Machine-Learning-Tokyo/CNN-Architectures,” GitHub. https://github.com/Machine-Learning-Tokyo/CNN-Architectures/tree/master/Implementations/Xception (Last accessed Aug. 26, 2023).

[4]“Softmax Activation Function: Everything You Need to Know,” Pinecone. https://www.pinecone.io/learn/softmax-activation/

[5]“Understanding Categorical Cross-Entropy Loss, Binary Cross-Entropy Loss, Softmax Loss, Logistic Loss, Focal Loss and all those confusing names,” Github.io, 2018. https://gombru.github.io/2018/05/23/cross_entropy_loss/

[6]“Glossary of Terms,” Machine Learning, vol. 30, no. 2, pp. 271–274, Feb. 1998, doi:https://doi.org/10.1023/A:1017181826899.

T-SNE IN SHORT

- Creates a probability distribution that measures pairwise similarity between points in higher dimensional space.
- Similar probability distribution is created for lower dimensional space.
- The goal is to position the points in lower dimensional space such that pairwise similarities of the higher dimension are preserved.
- t-SNE optimizes the points in lower dimensional space by minimizing the divergence between the two probability distributions. This is an iterative process done by gradient descend.

UMAP IN SHORT

- creates a graph in higher dimension space that captures the local relationship between points in higher dimensional space.
- Creates a fuzzy topological structure by measuring the similarity between the points.
- optimizes the points in lower dimensional space to approximate this structure.
- again the aim is to minimize the difference between the point-wise similarity between the points in higher dimensional space.