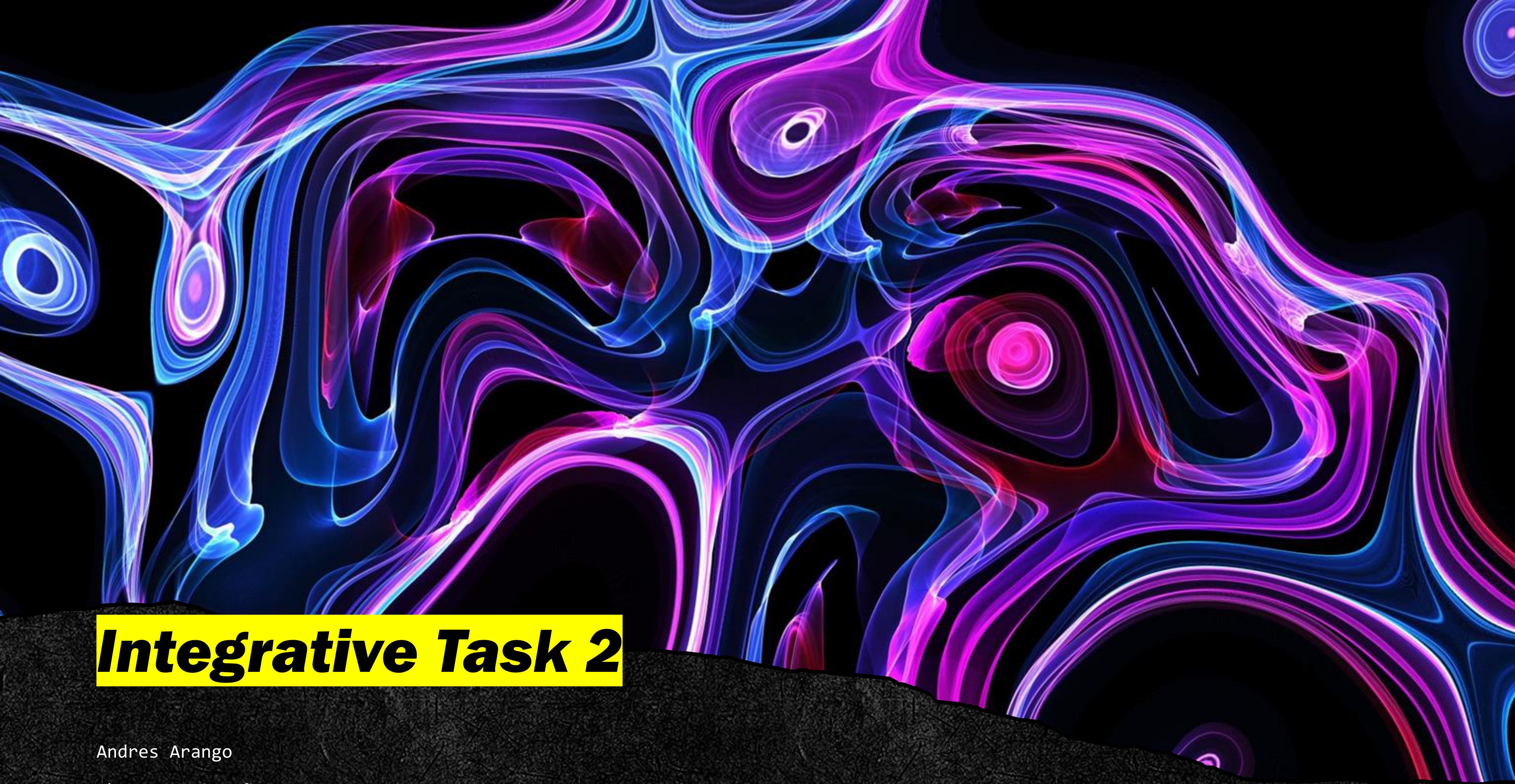


Integrative Task 2

Andres Arango

Jhonatan Castaño

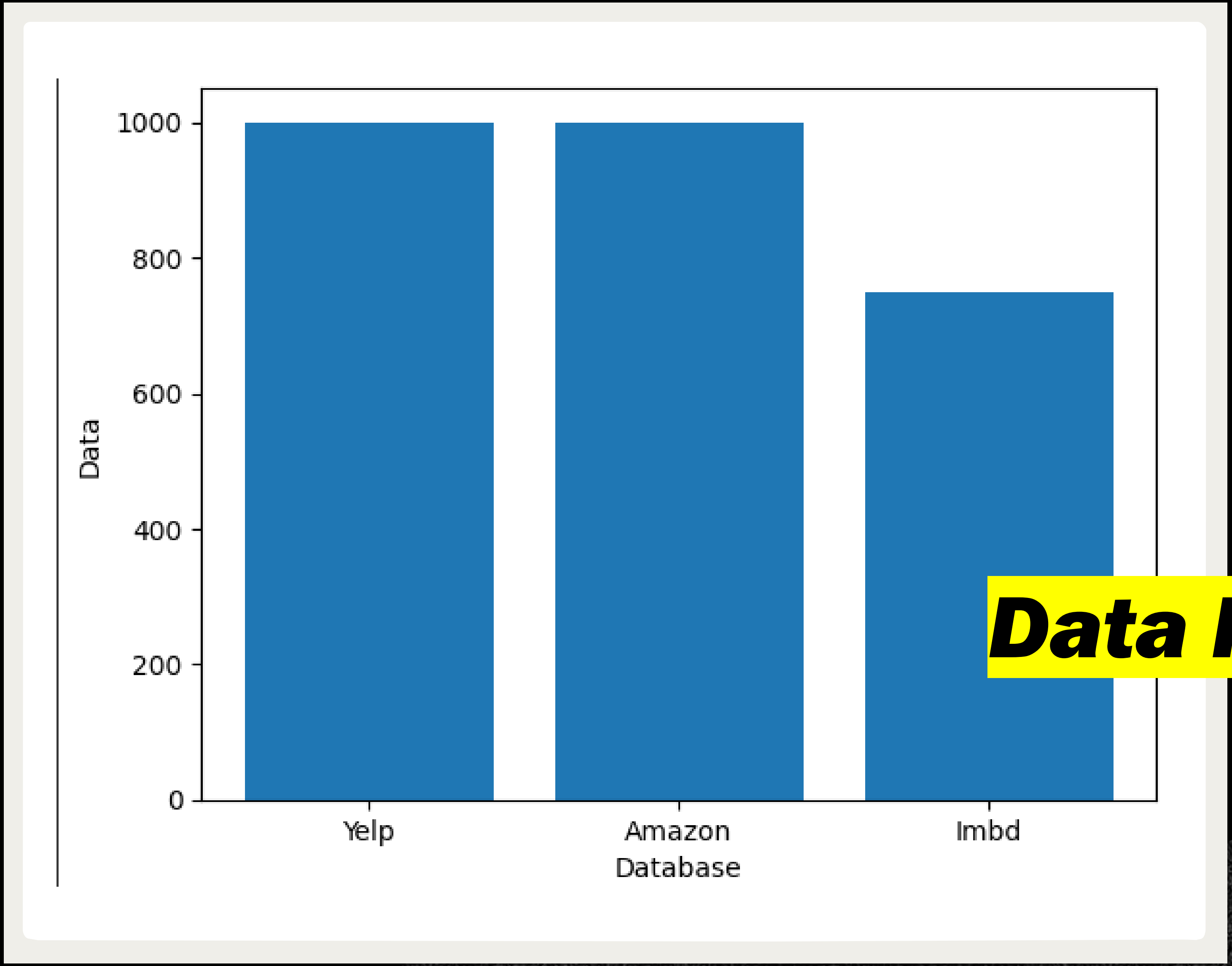
Introduction

- Objective: Build a sentiment analysis model using supervised learning with vanilla Recurrent Neural Networks and LSTM
- Secondary Objectives:

1. Create a database with sentences and the type of sentiment of itself.
2. Tokenize the sentences to find a way to build a supervised learning model.
3. Implement a DummyClassifier for the model.
4. Implement a vanilla RNN sentiment analysis model.
5. Implement a LSTM sentiment analysis model.

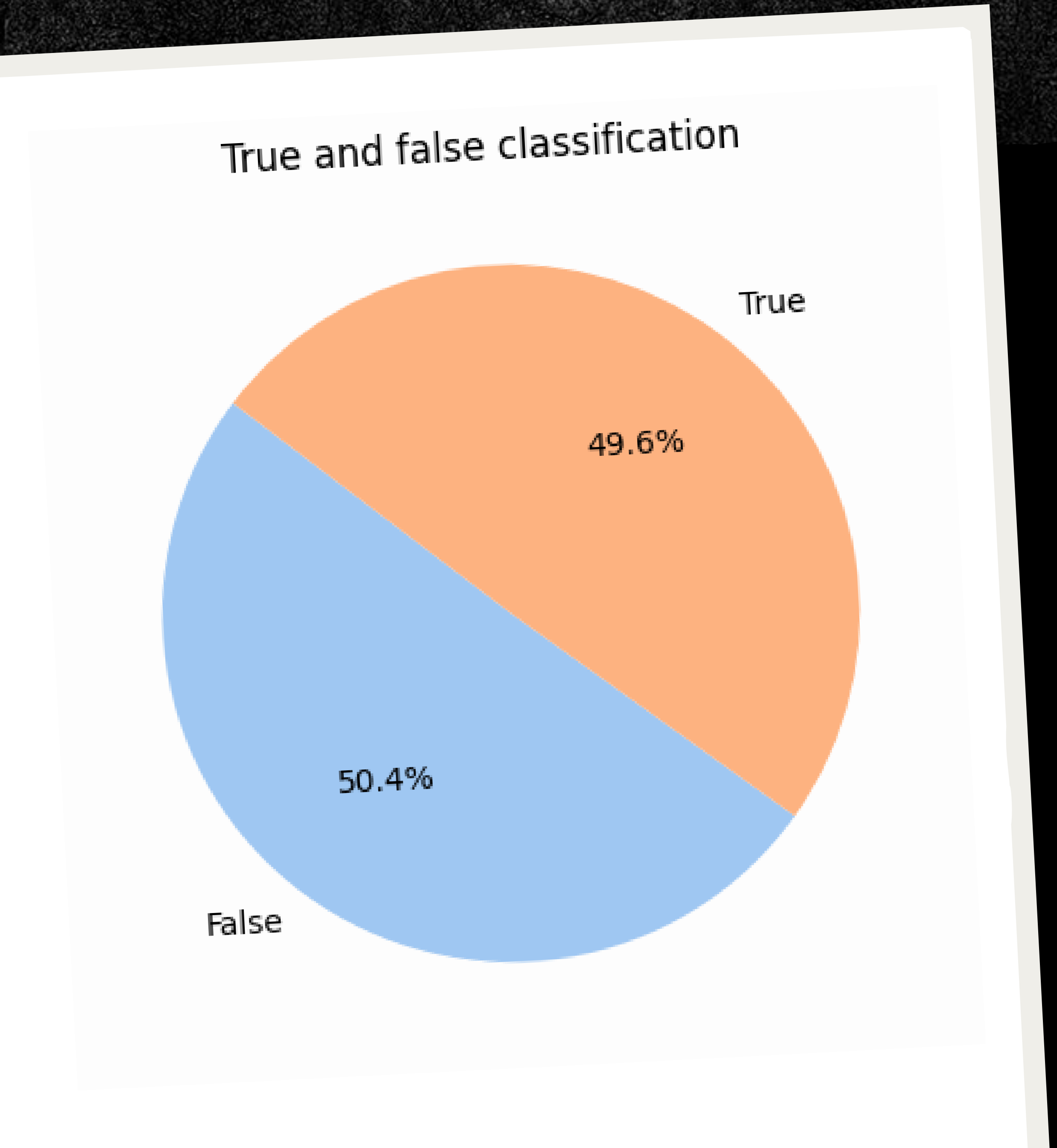
Methodology

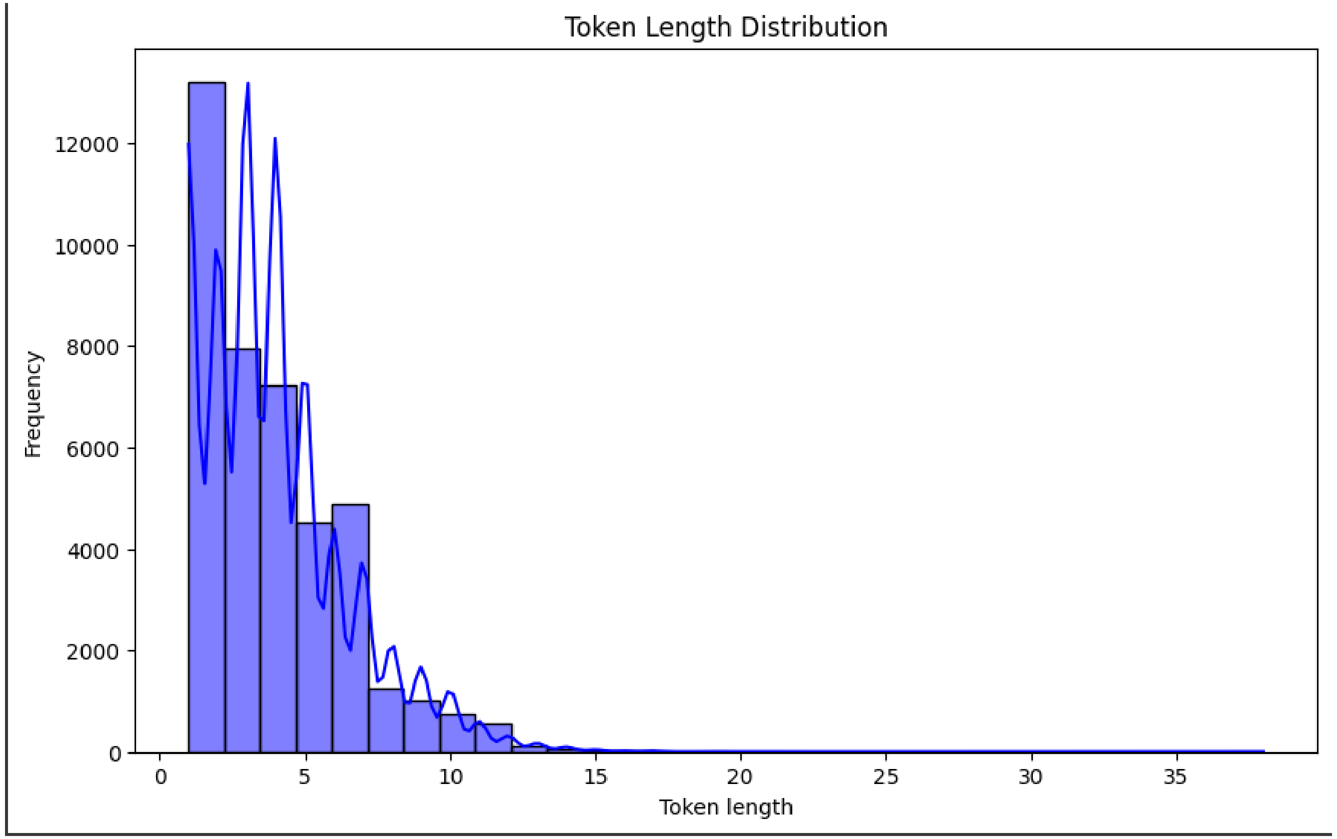
- Define the problem
- Gather Data
- Explore and prepare data
- Feature Engineerin
- Split the Data
- Choose a Model
- Train the Model
- Evaluate Model
- Hyperparameter Tuning



Data From Each of the Databases

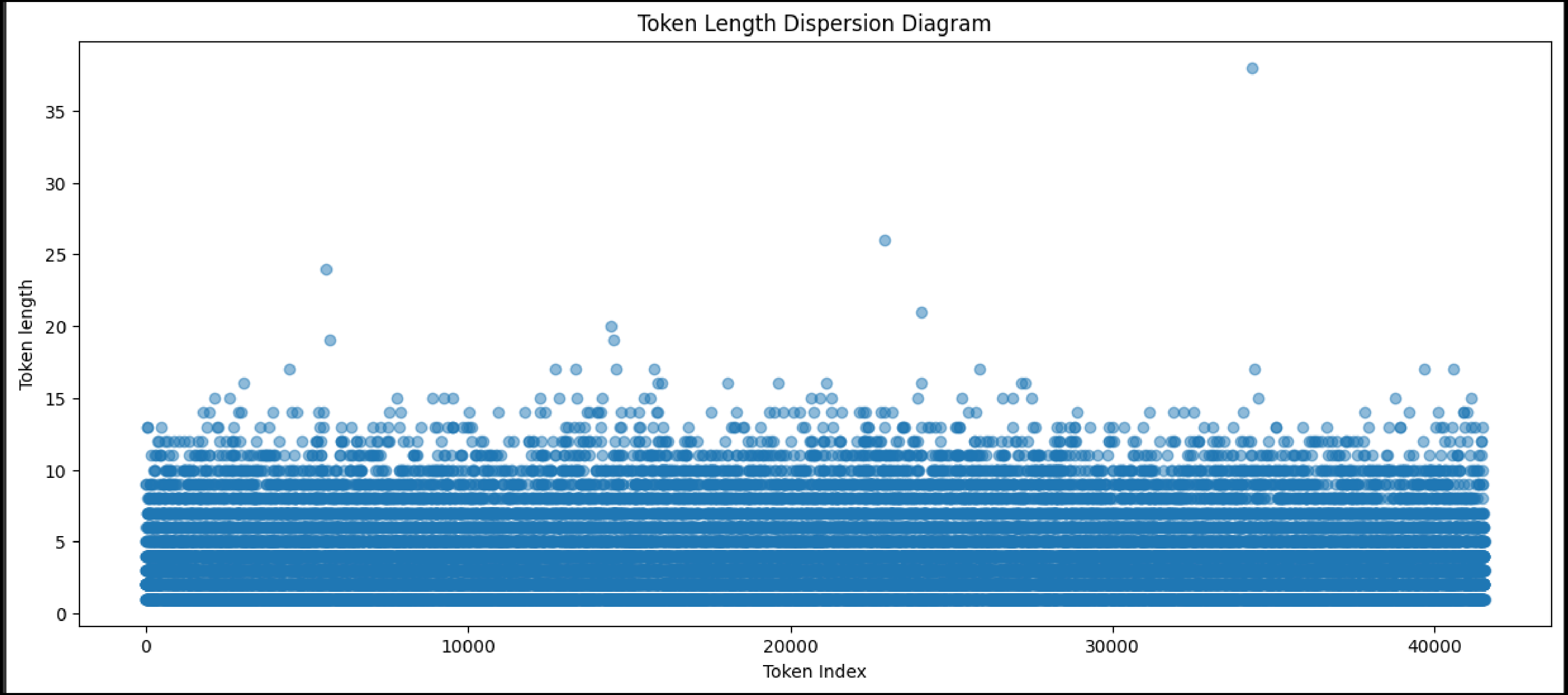
Clasiffication



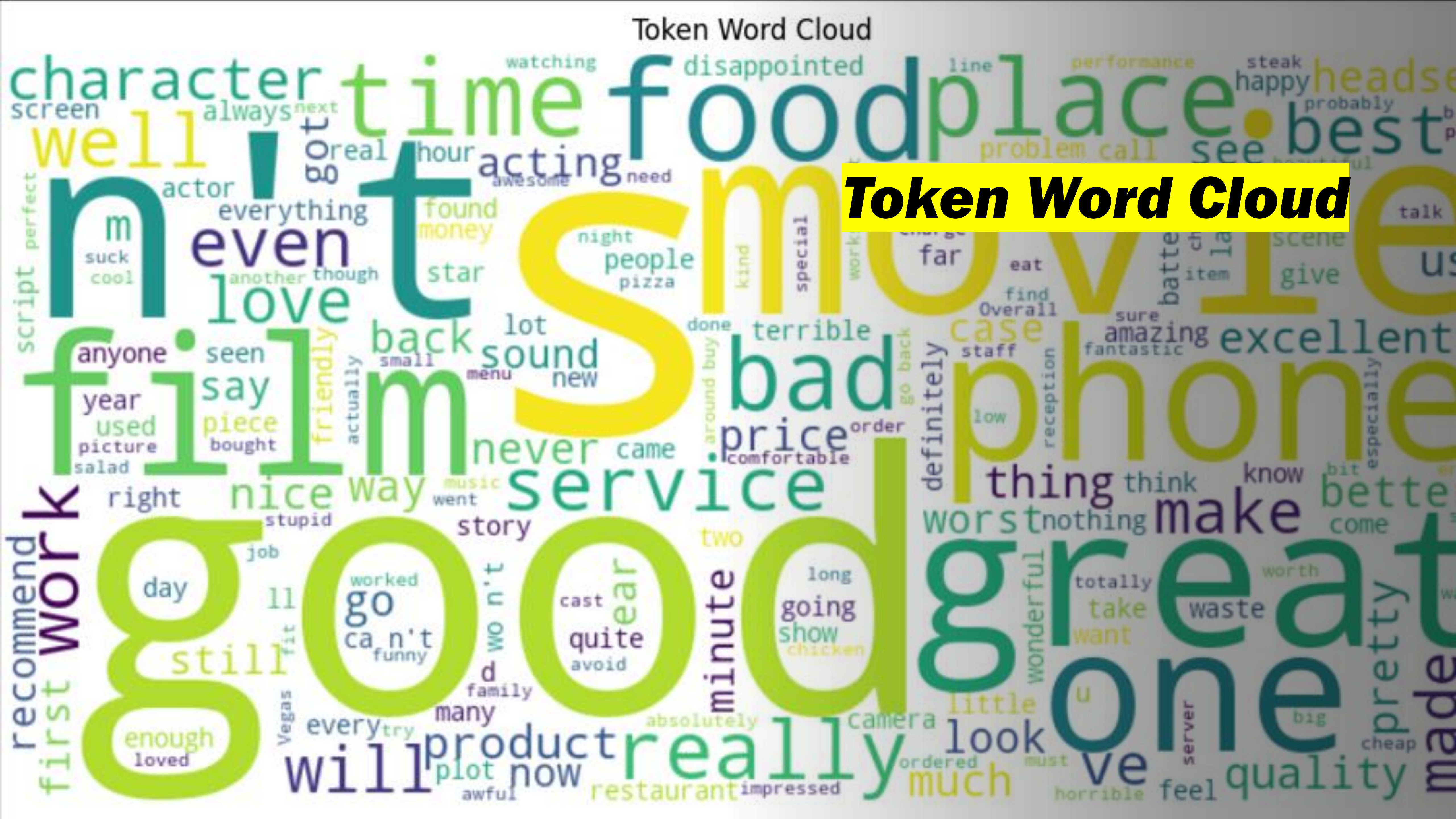


Token length Distribution

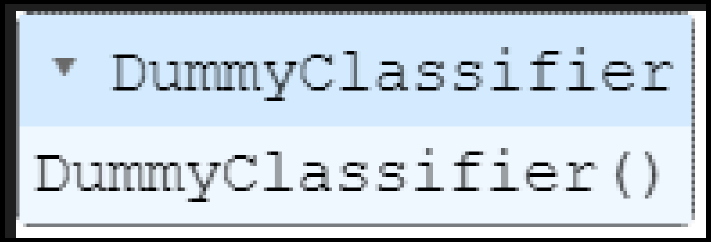
Token length Dispersion



Token Word Cloud

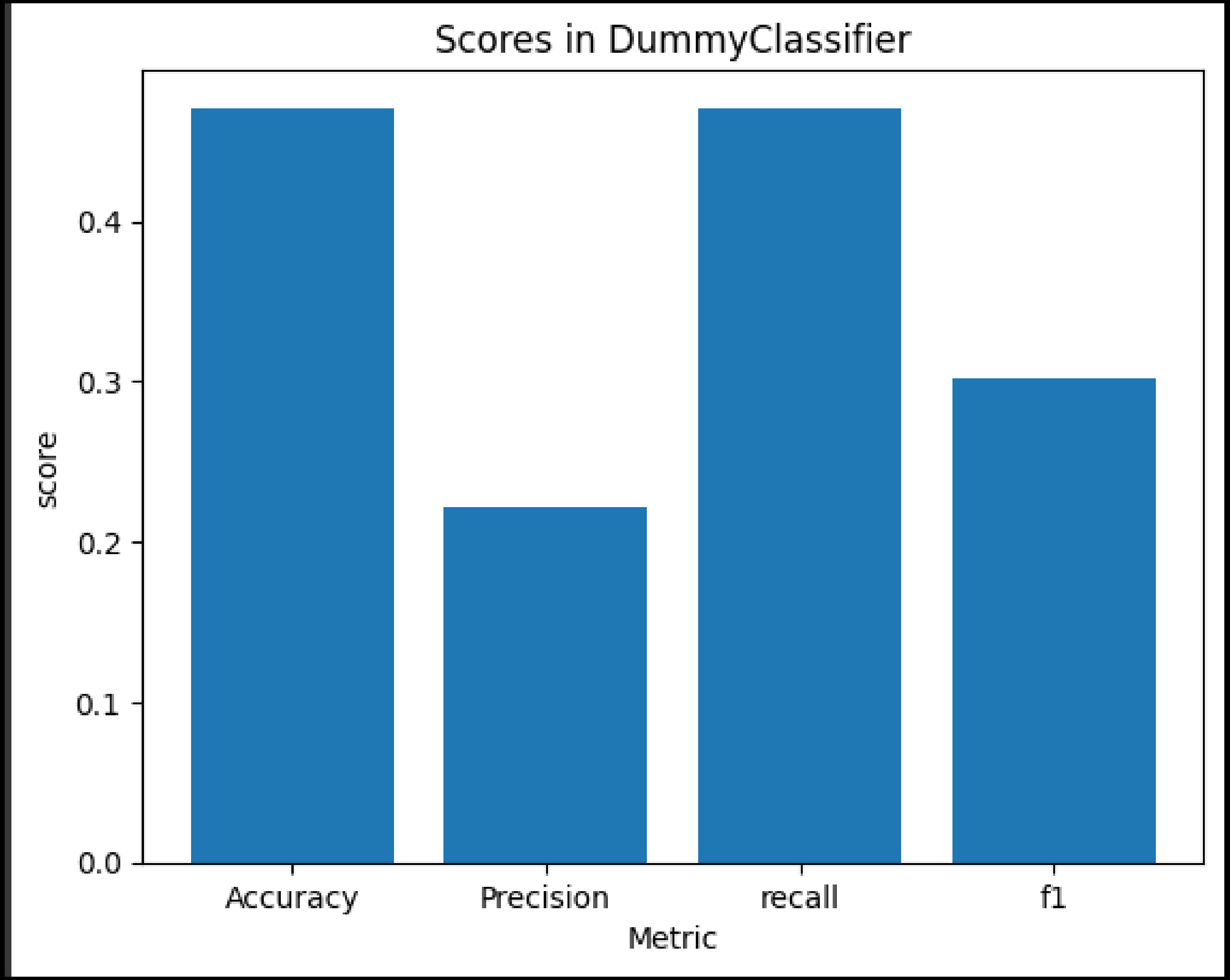


Dummy Classifier Implementation

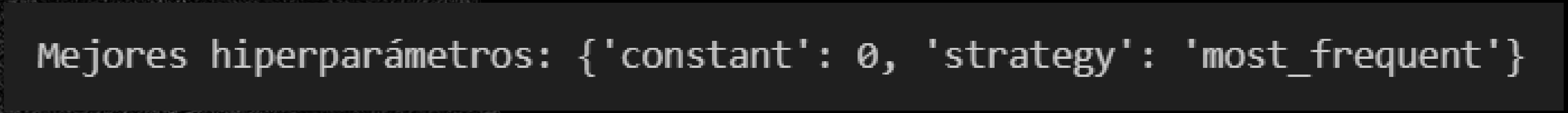


```
▾ DummyClassifier
DummyClassifier()
```

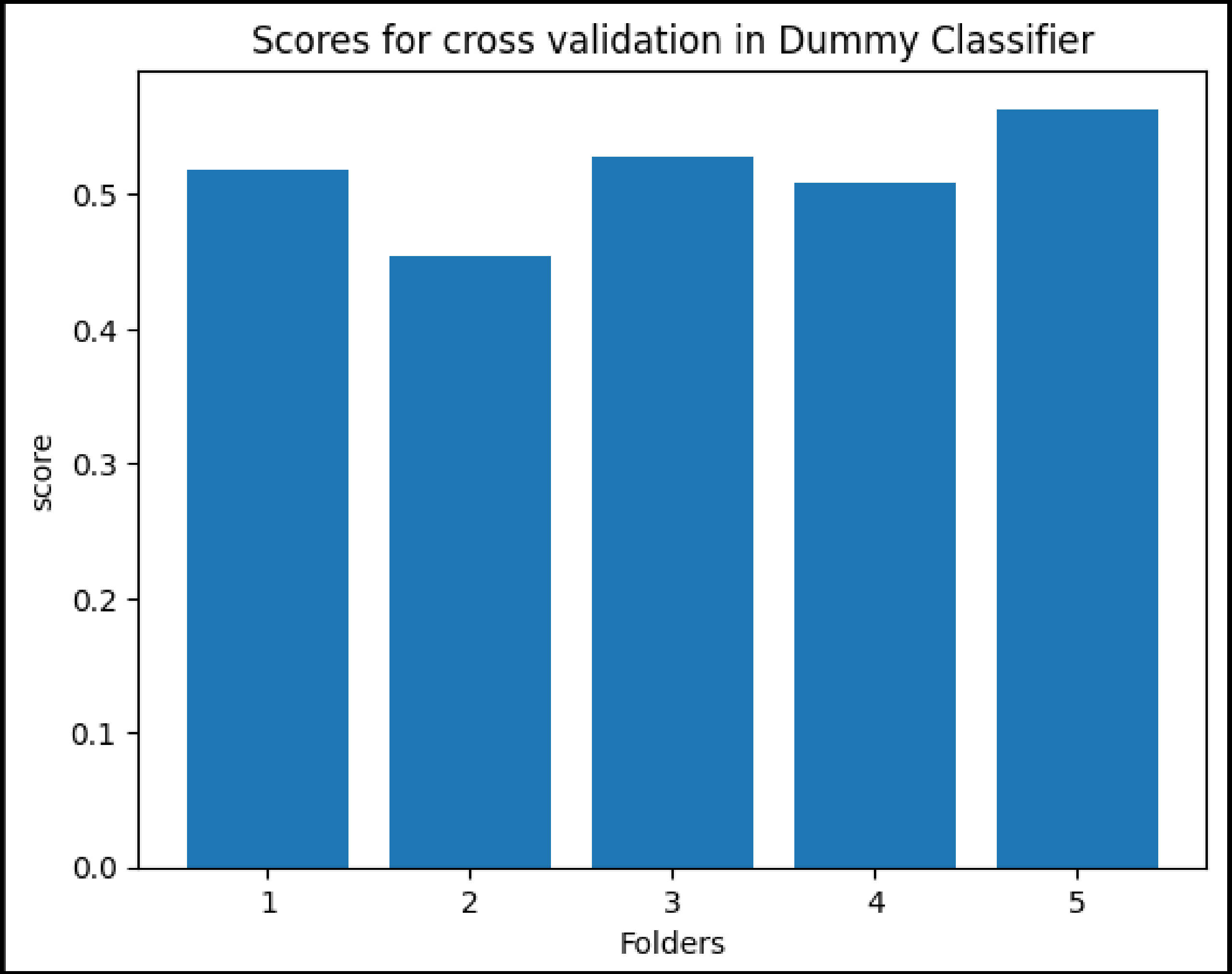
Scores in DummyClassifier



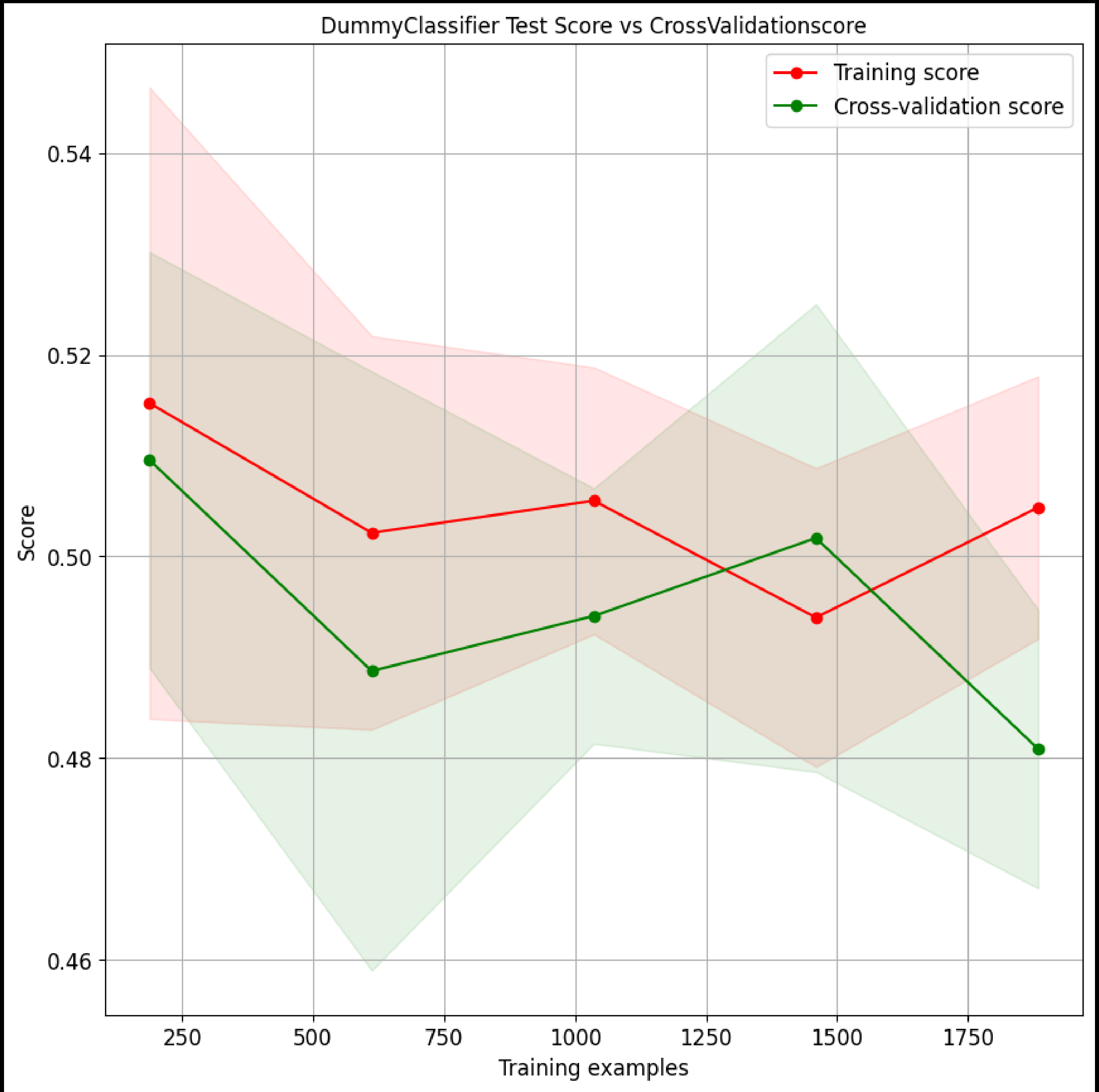
DummyClassifier Hyperparameter Tunning



```
Mejores hiperparámetros: {'constant': 0, 'strategy': 'most_frequent'}
```

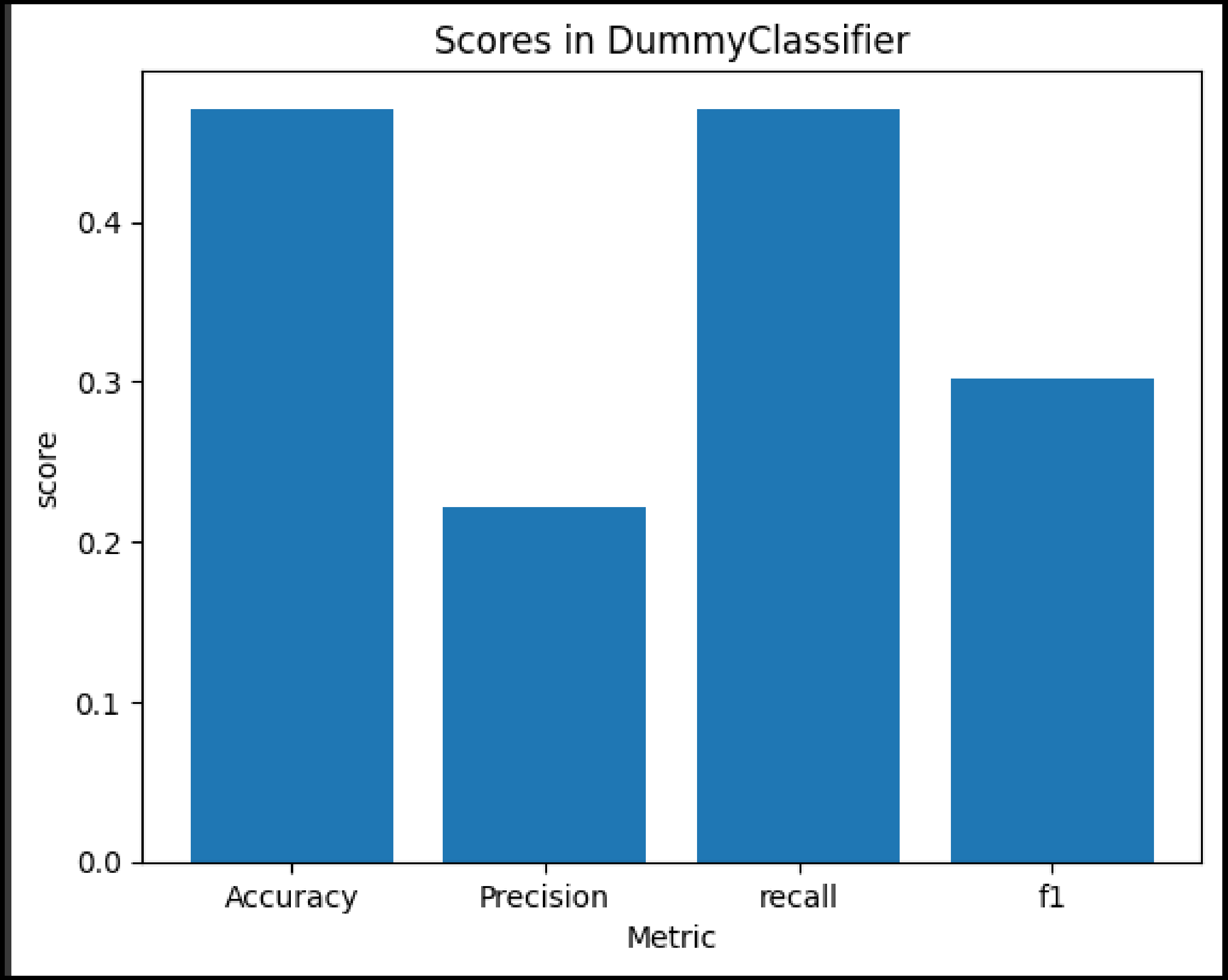


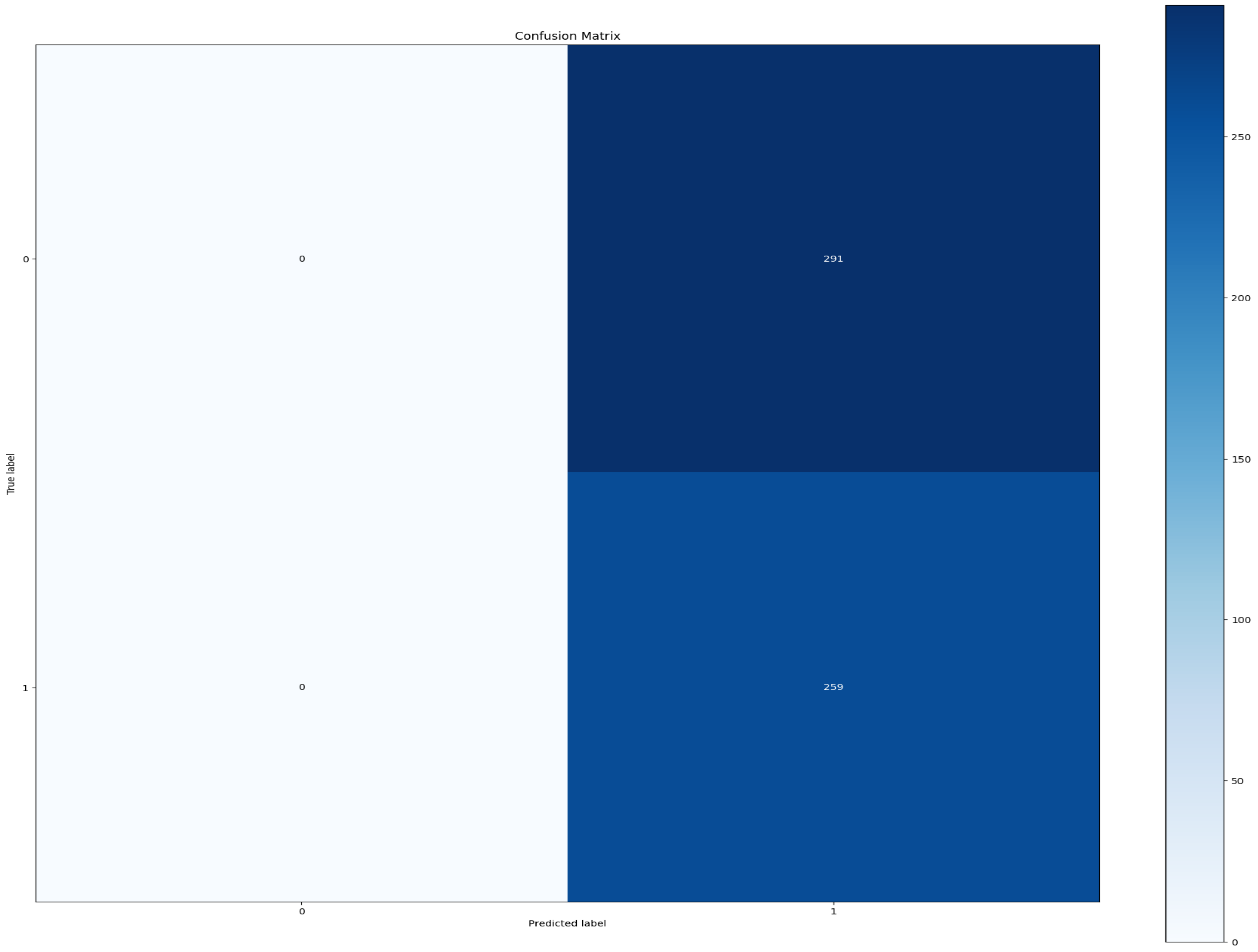
Scores for cross validation in DummyClassifier



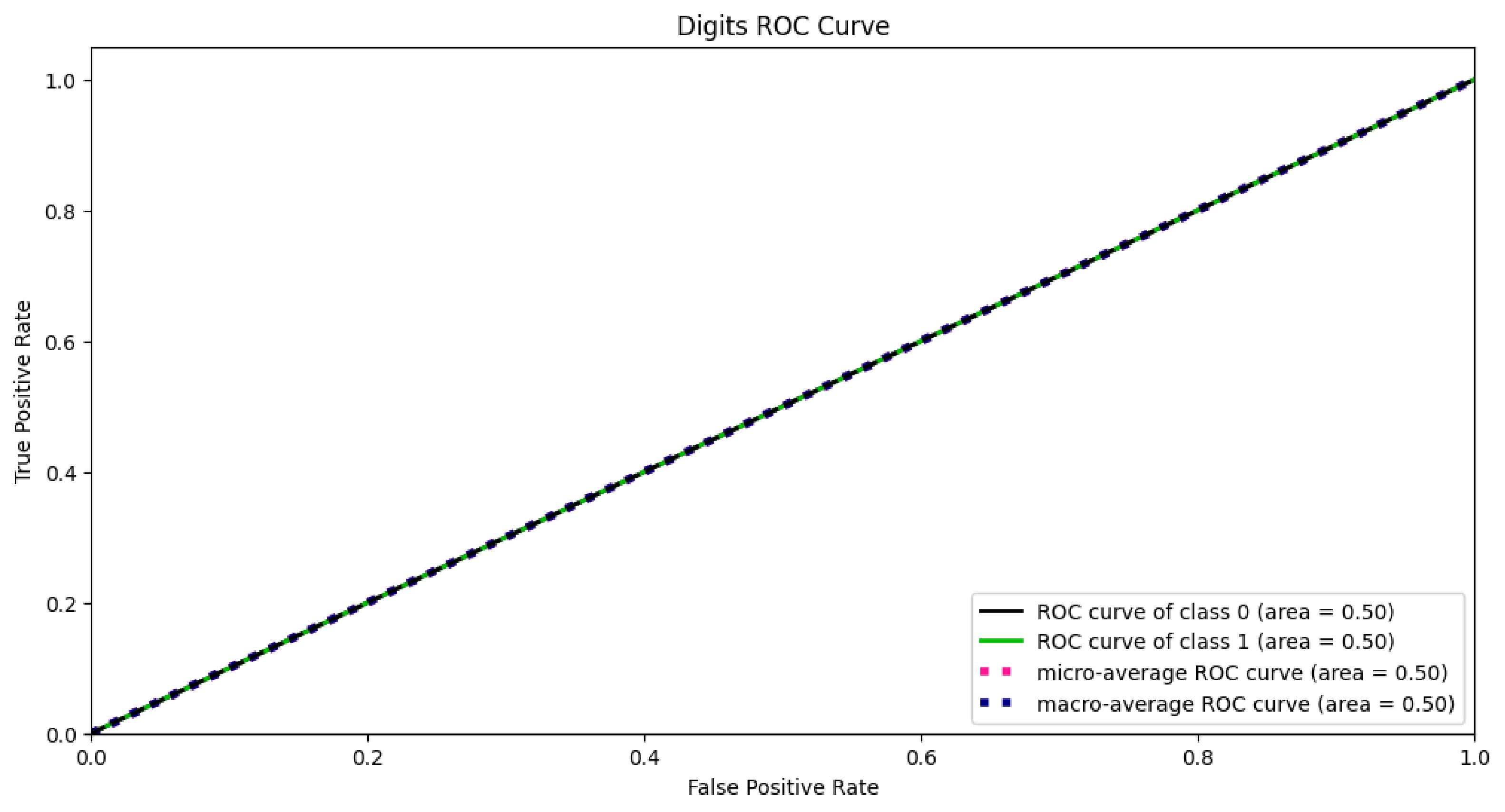
Cross validation vs Training results

Scores in DummyClassifier With DummyClassifier

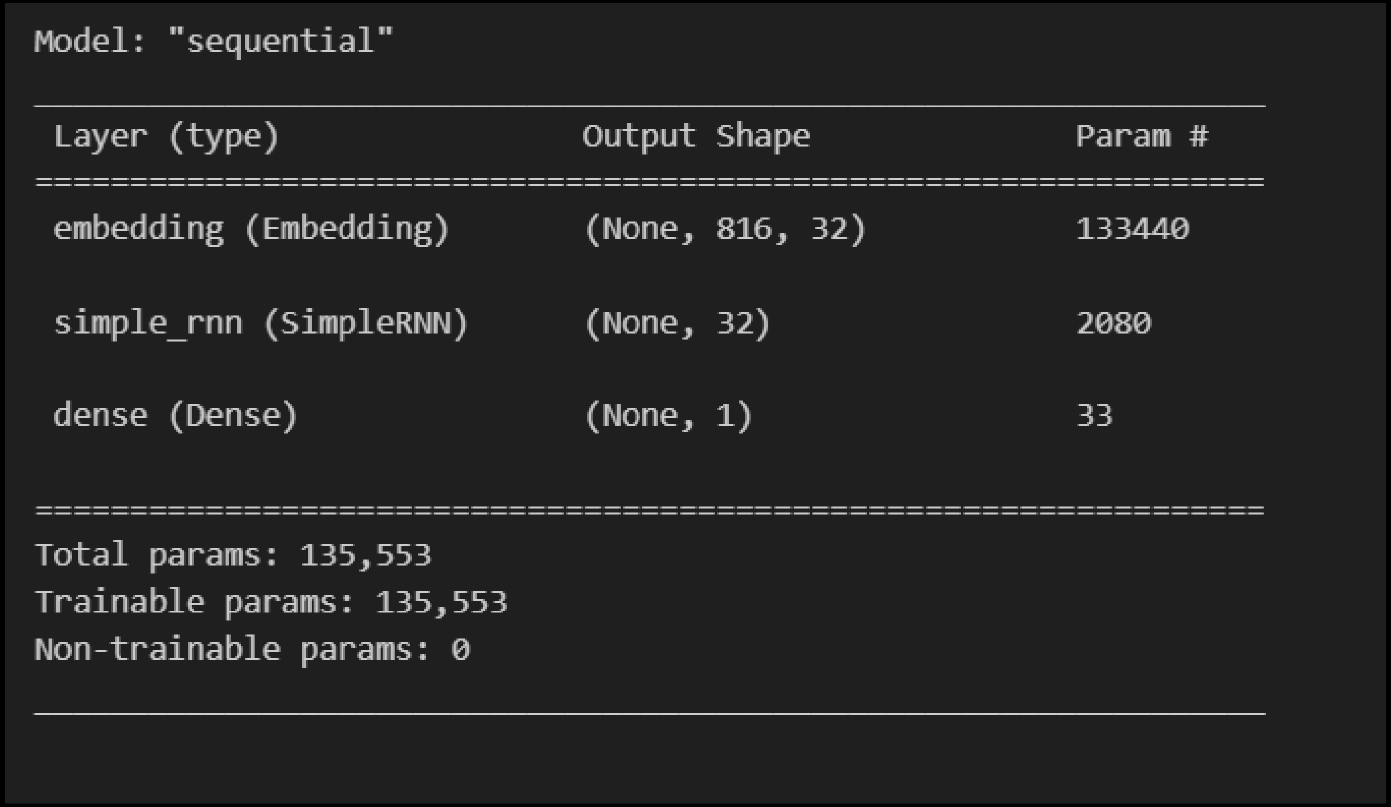



Confusion Matrix
True label
0
1
0
291
0
259
Predicted label
0
1
250
200
150
100
50
0

ROC curve

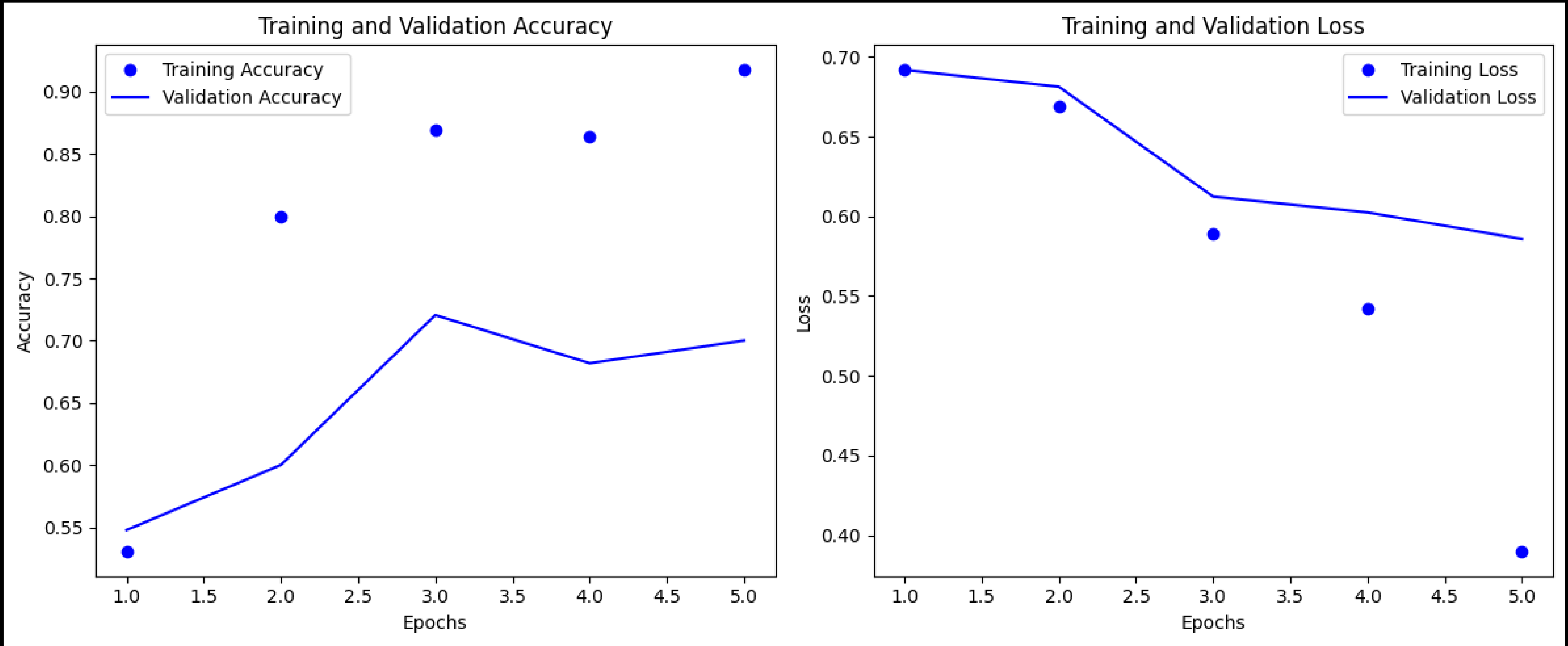

Digits ROC Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve of class 0 (area = 0.50)
ROC curve of class 1 (area = 0.50)
micro-average ROC curve (area = 0.50)
macro-average ROC curve (area = 0.50)

RNN Model

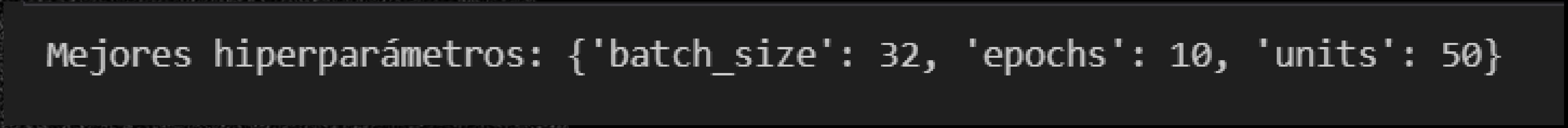


```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 816, 32)           133440

 simple_rnn (SimpleRNN)      (None, 32)                2080

 dense (Dense)               (None, 1)                 33

=================================================================
Total params: 135,553
Trainable params: 135,553
Non-trainable params: 0
_________________________________________________________________
```

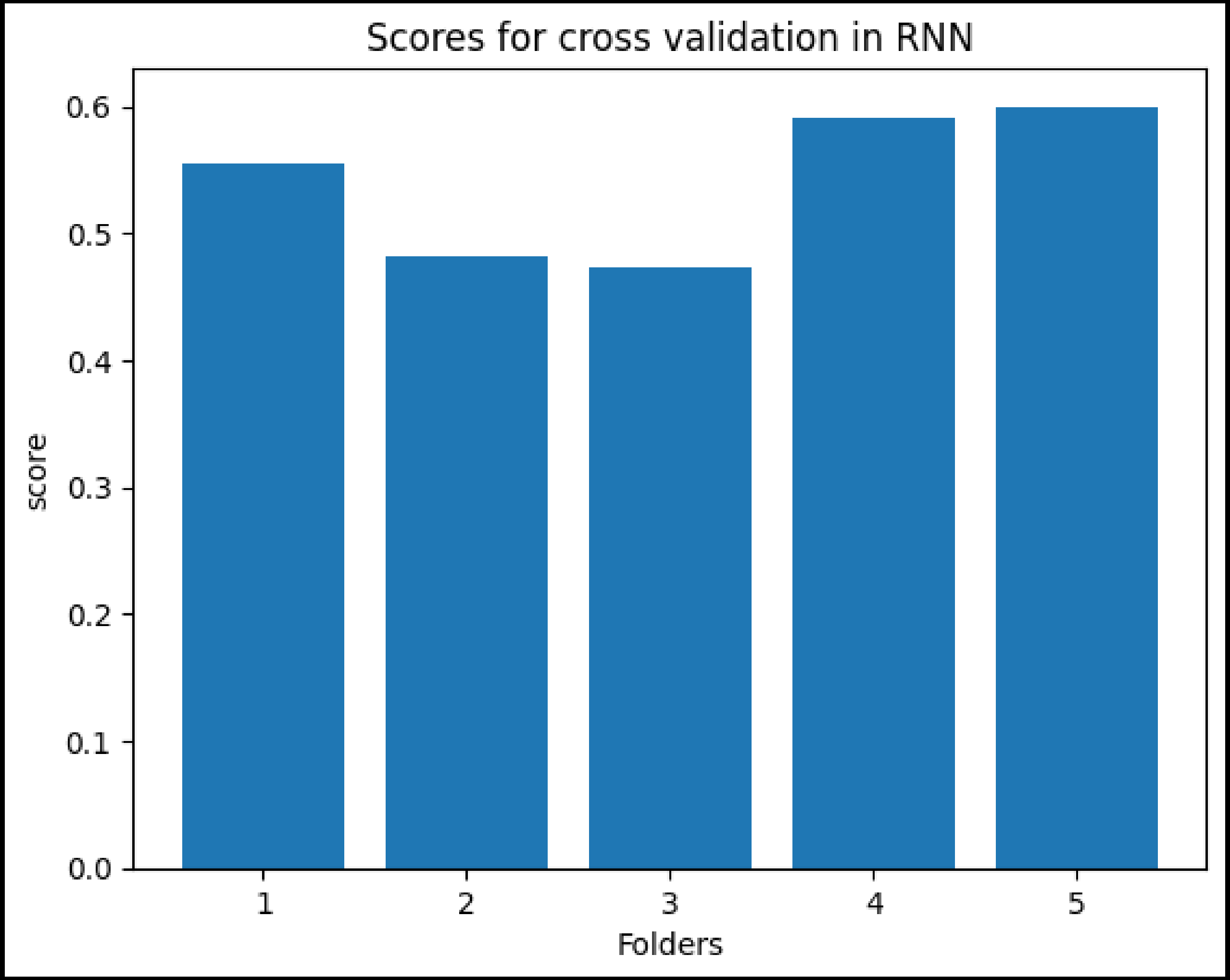
RNN Model Evaluation



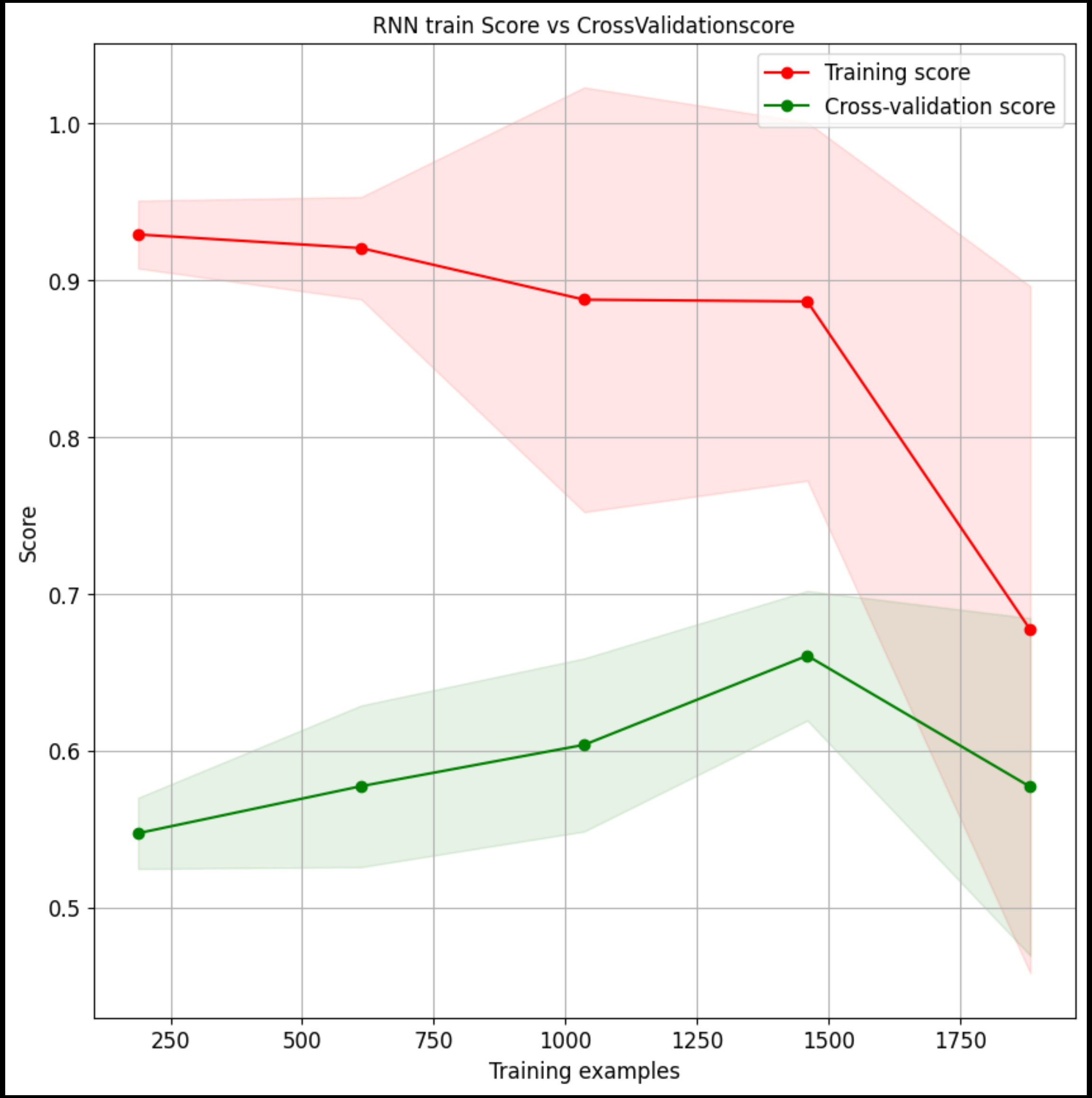
RNN Hyperparameter Tunning



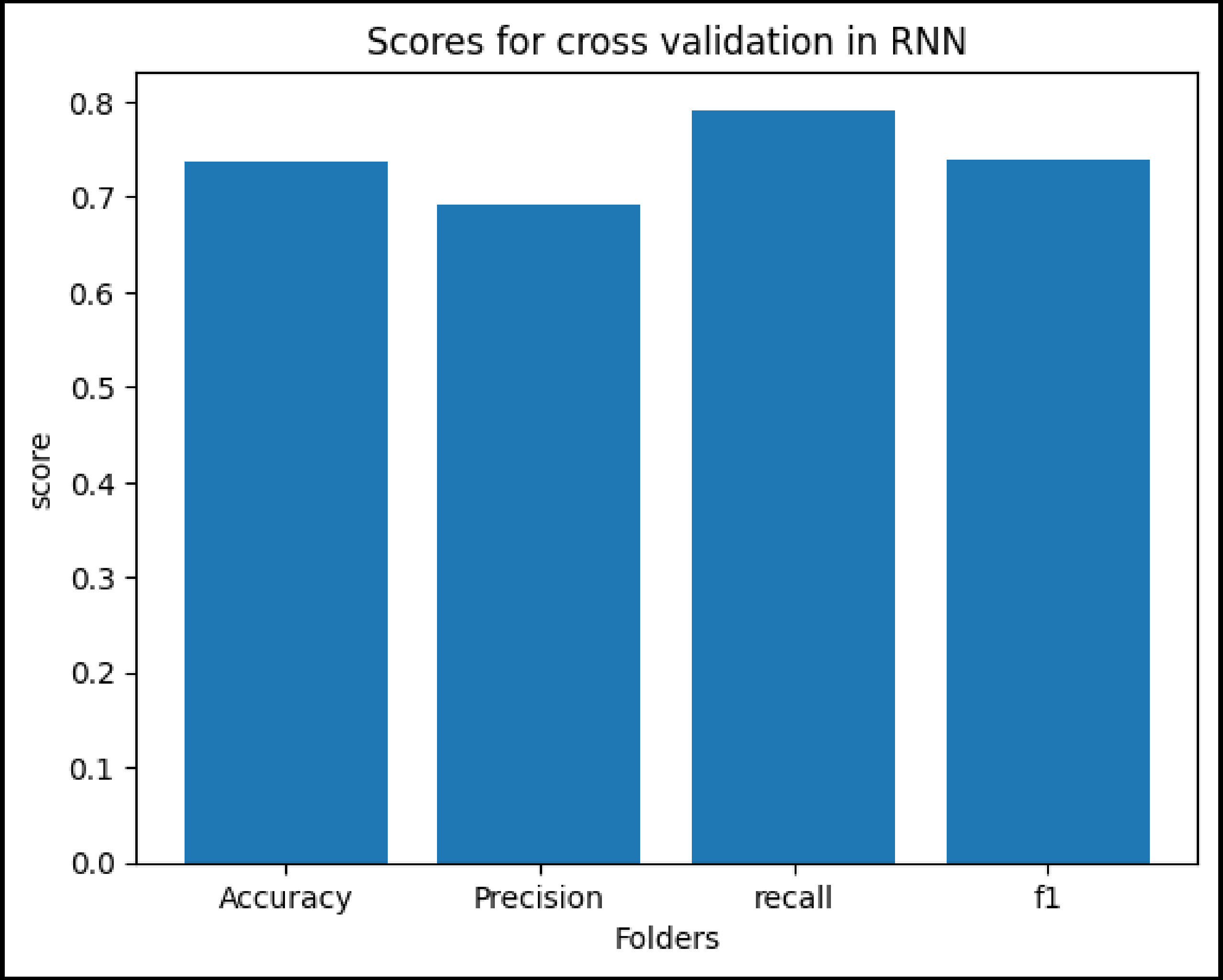
```
Mejores hiperparámetros: {'batch_size': 32, 'epochs': 10, 'units': 50}
```



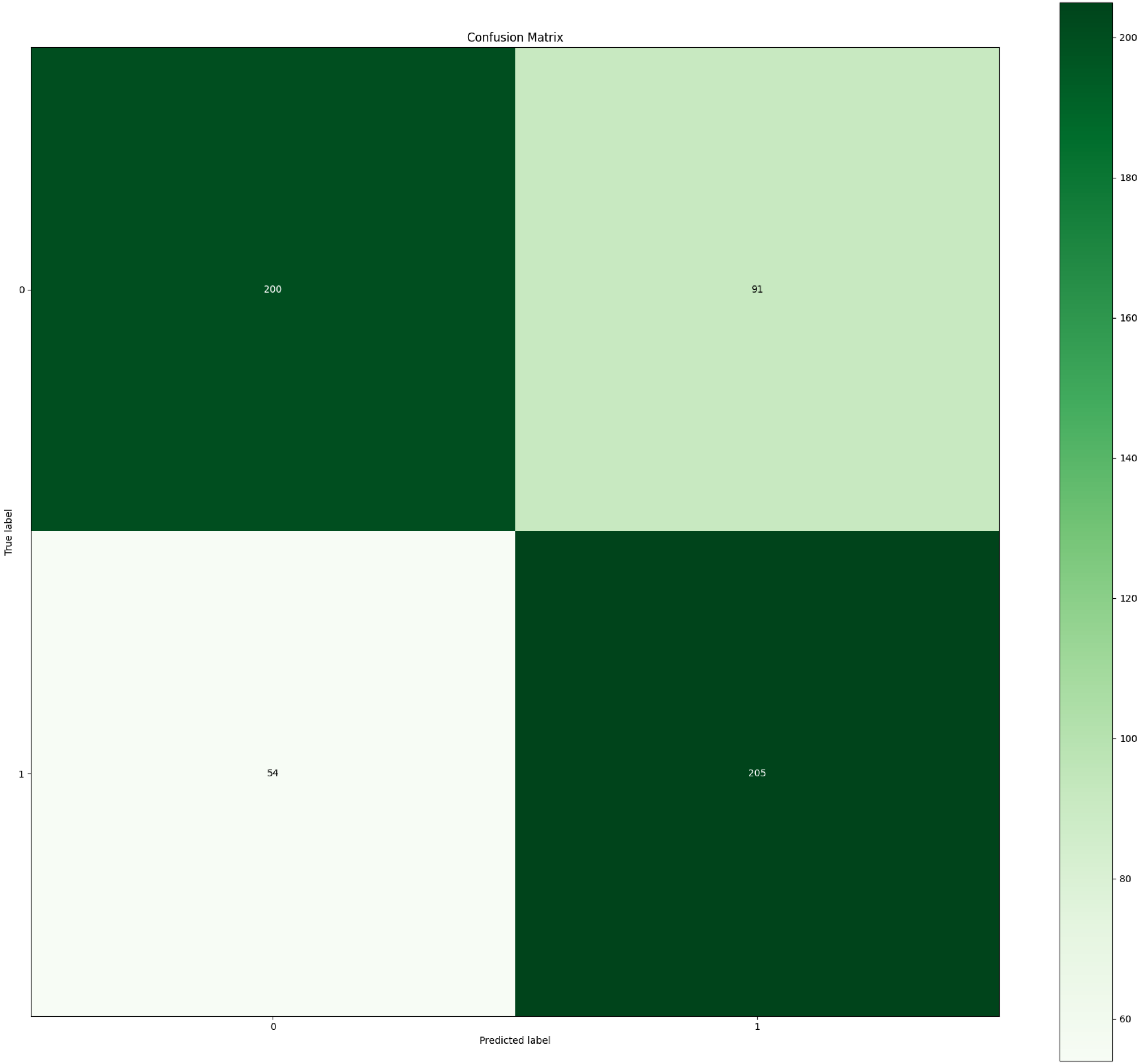
RNN Cross validation



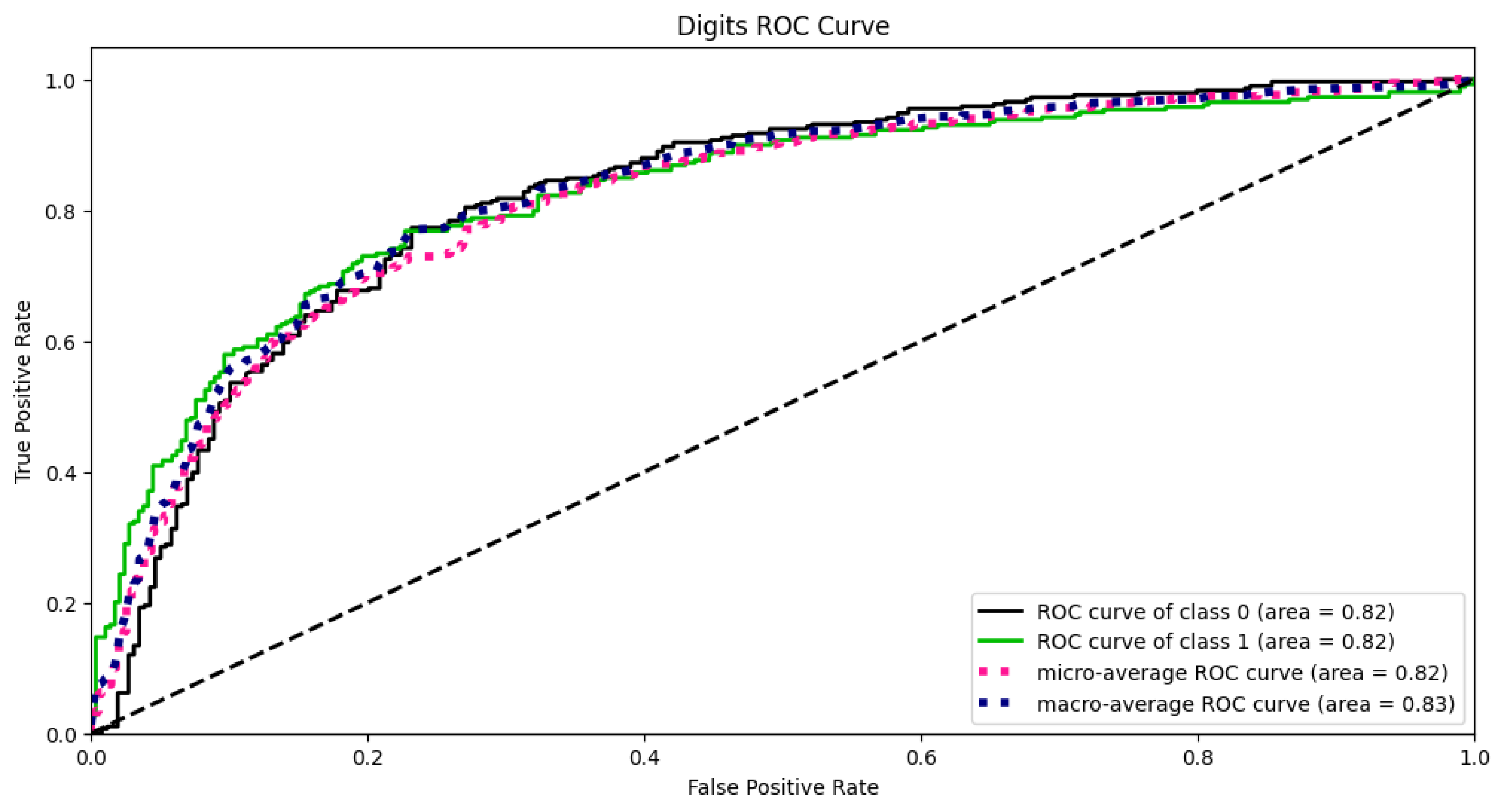
RNN Train Score vs Cross Validation



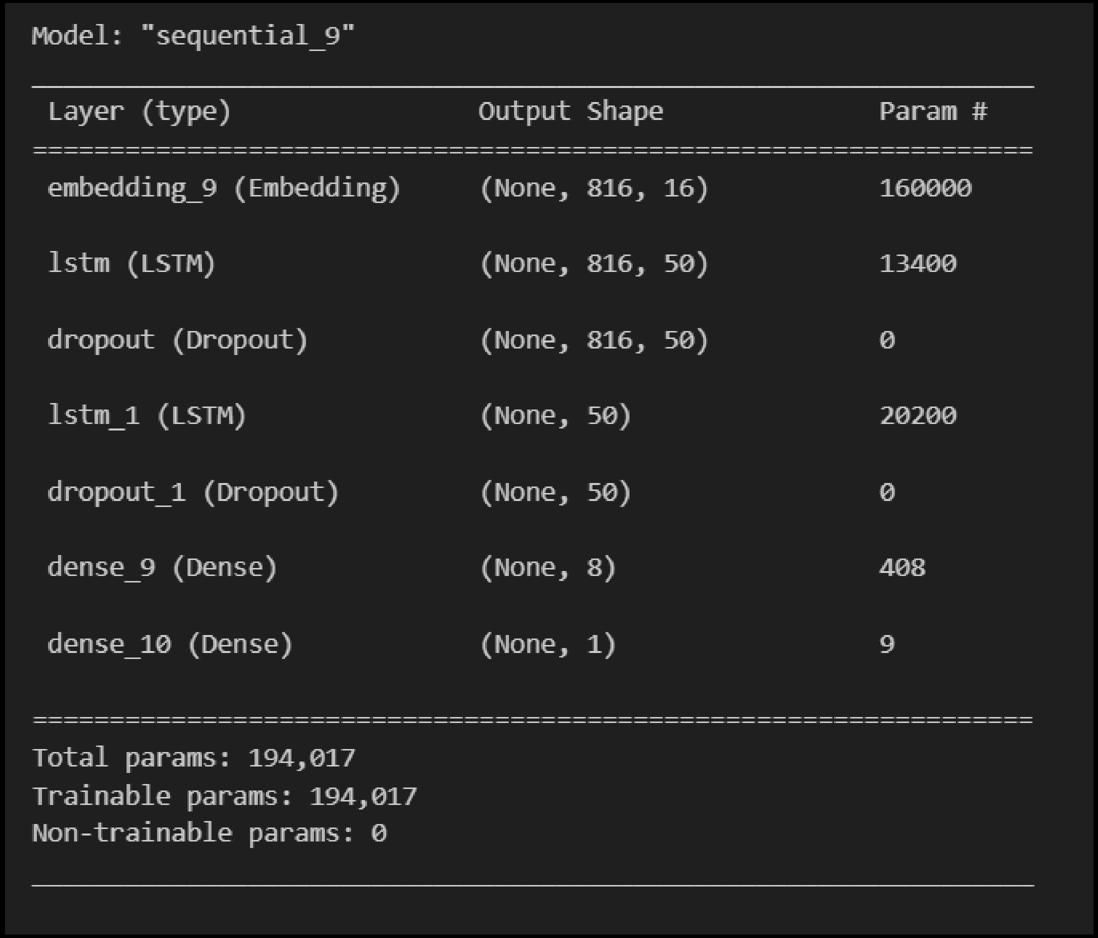
RNN Cross validation


Confusion Matrix
True label
0
1
200
91
54
205
0
1
Predicted label
200
180
160
140
120
100
80
60

ROC curve

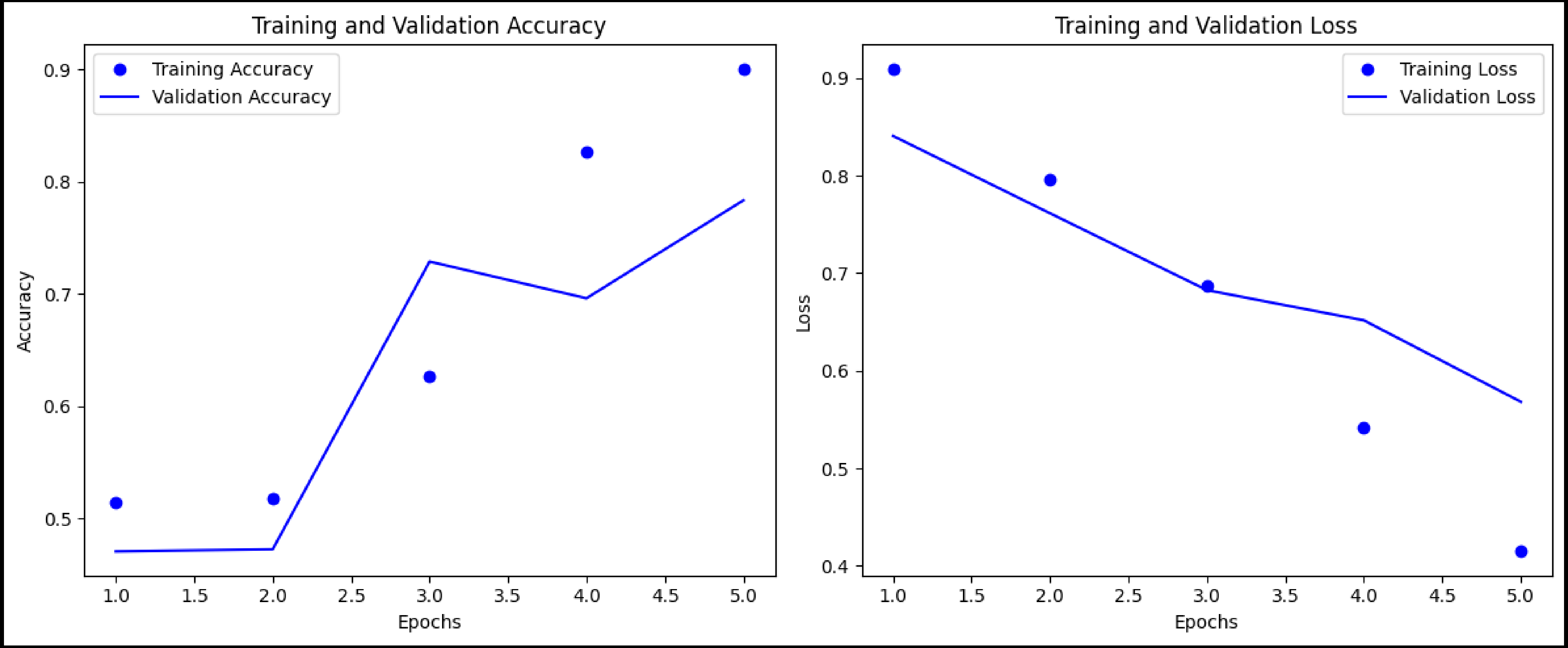

Digits ROC Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve of class 0 (area = 0.82)
ROC curve of class 1 (area = 0.82)
micro-average ROC curve (area = 0.82)
macro-average ROC curve (area = 0.83)

LSMT Model

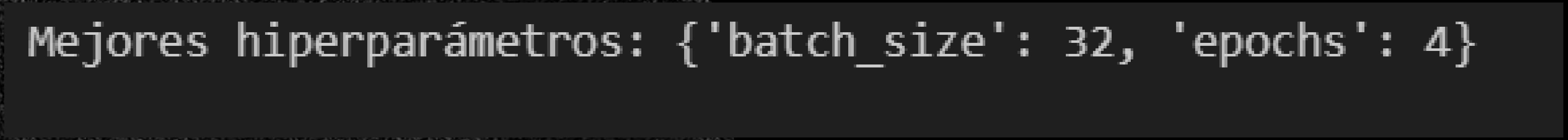


```
Model: "sequential_9"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_9 (Embedding)     (None, 816, 16)           160000

 lstm (LSTM)                 (None, 816, 50)           13400

 dropout (Dropout)           (None, 816, 50)           0

 lstm_1 (LSTM)               (None, 50)                20200

 dropout_1 (Dropout)         (None, 50)                0

 dense_9 (Dense)             (None, 8)                 408

 dense_10 (Dense)            (None, 1)                 9

=================================================================
Total params: 194,017
Trainable params: 194,017
Non-trainable params: 0
_________________________________________________________________
```

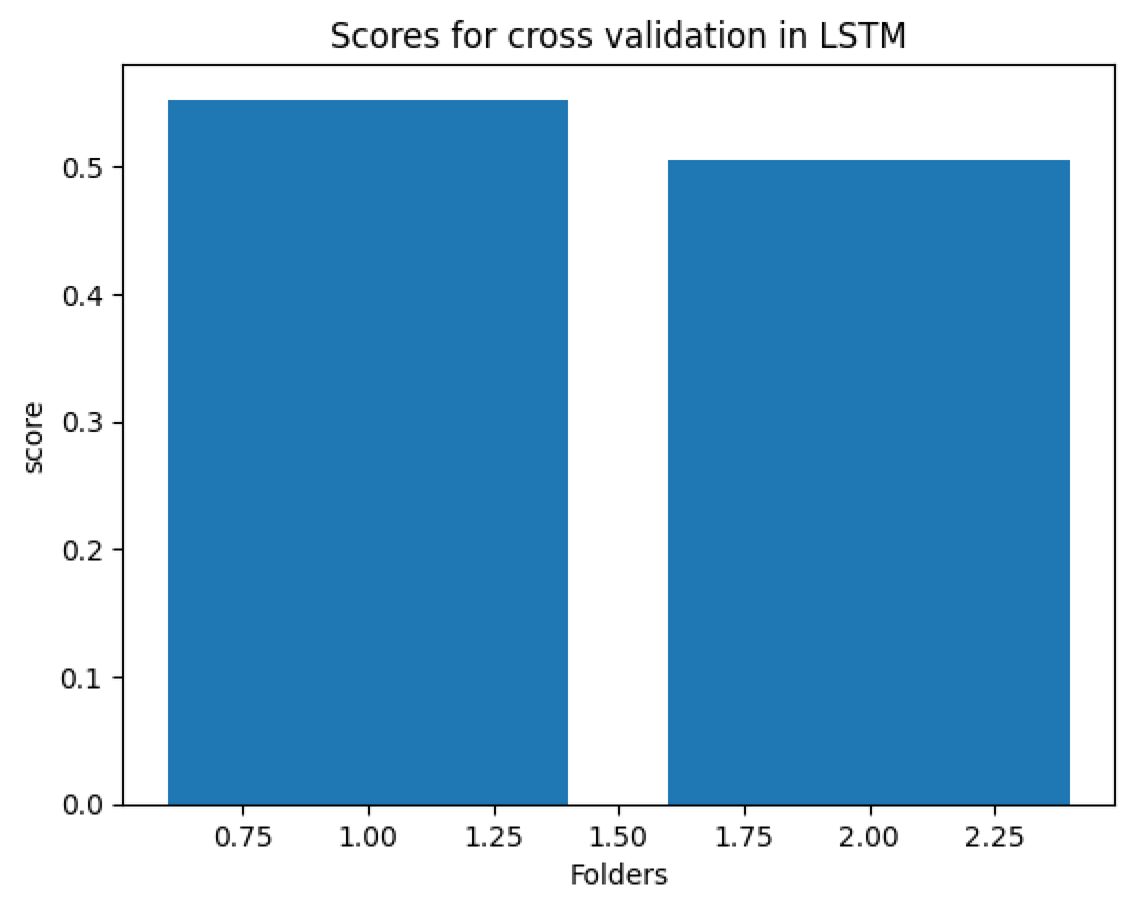
RNN Model Evaluation



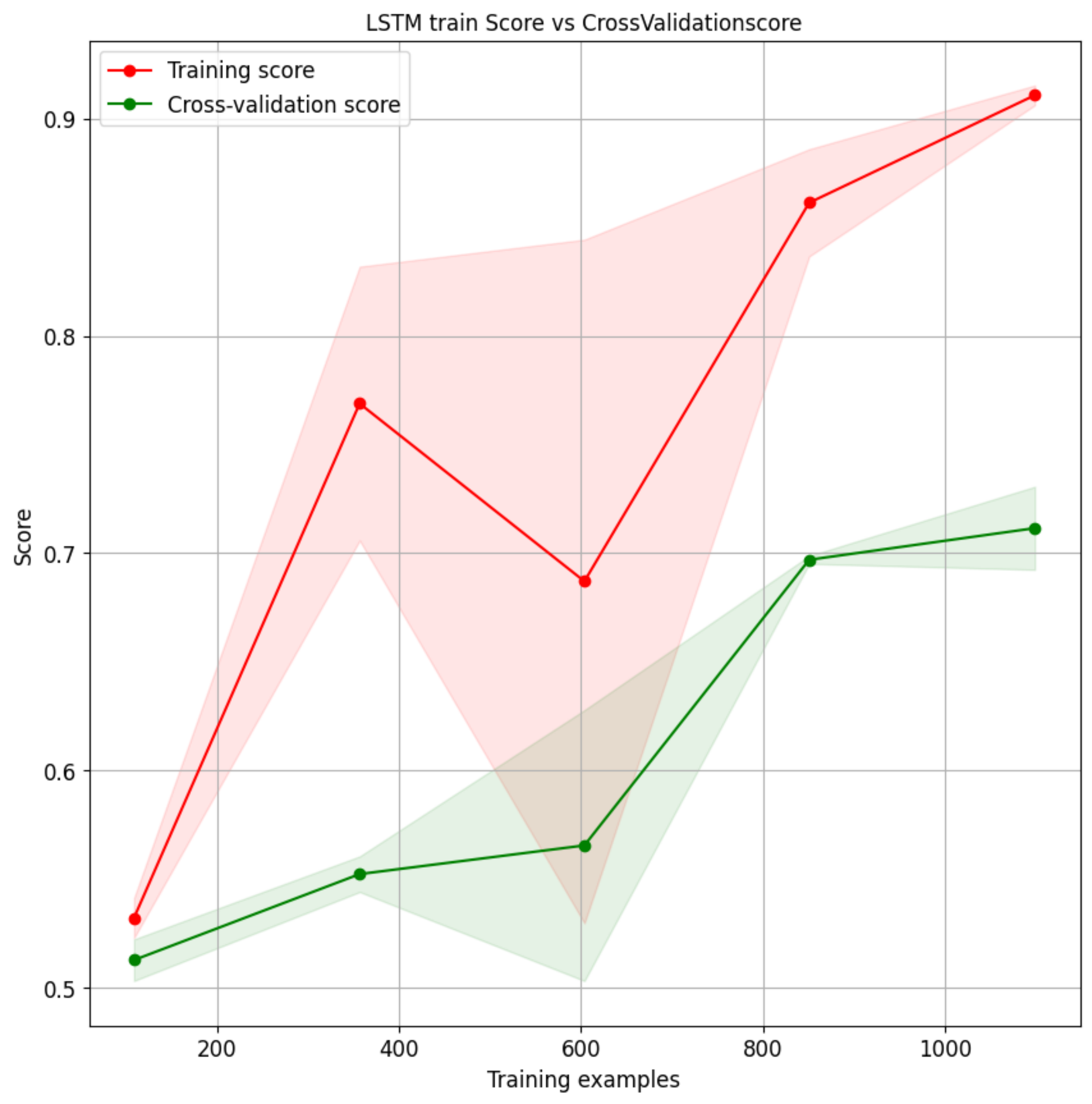
LSTM Hyperparameter Tunning



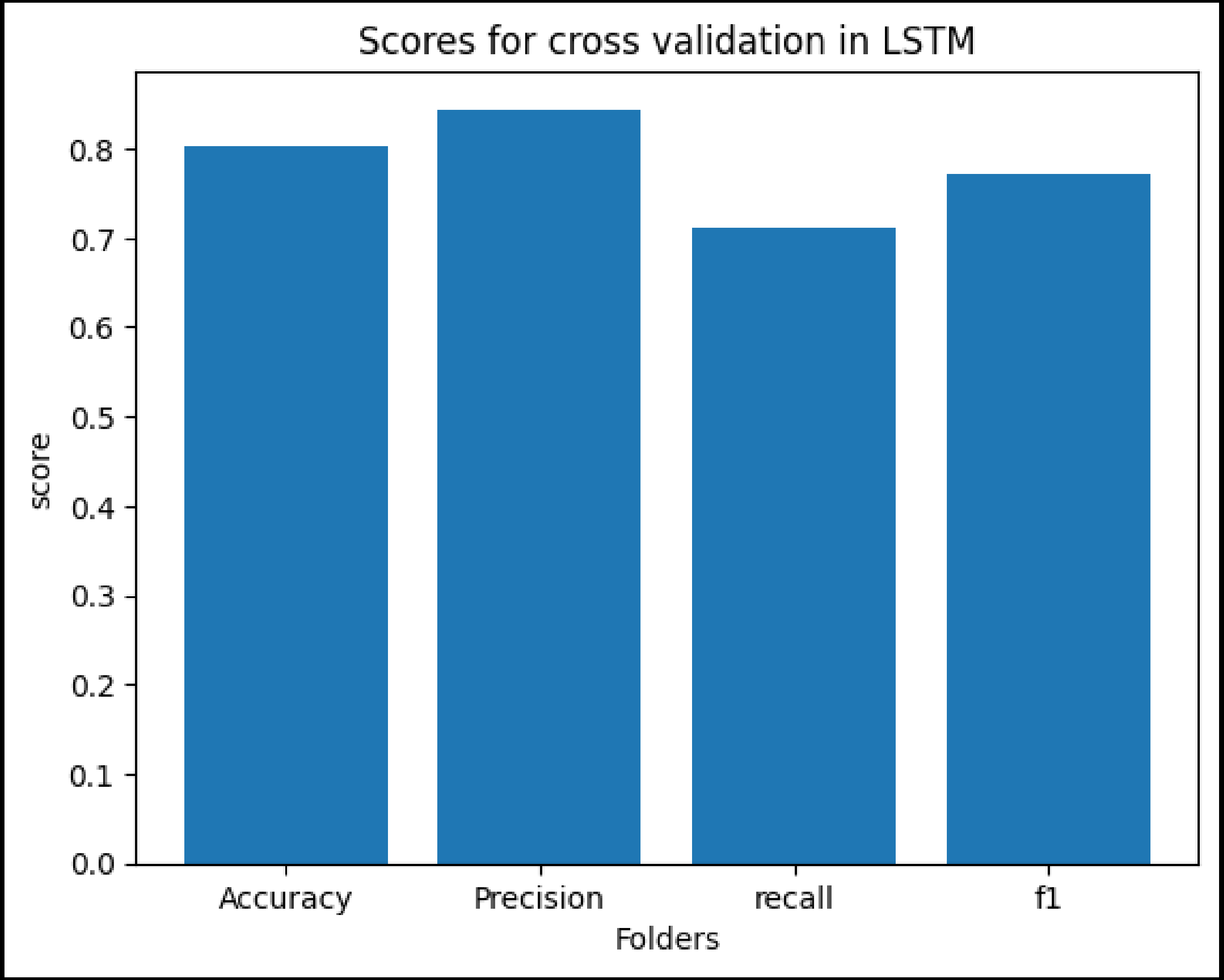
```
Mejores hiperparámetros: {'batch_size': 32, 'epochs': 4}
```



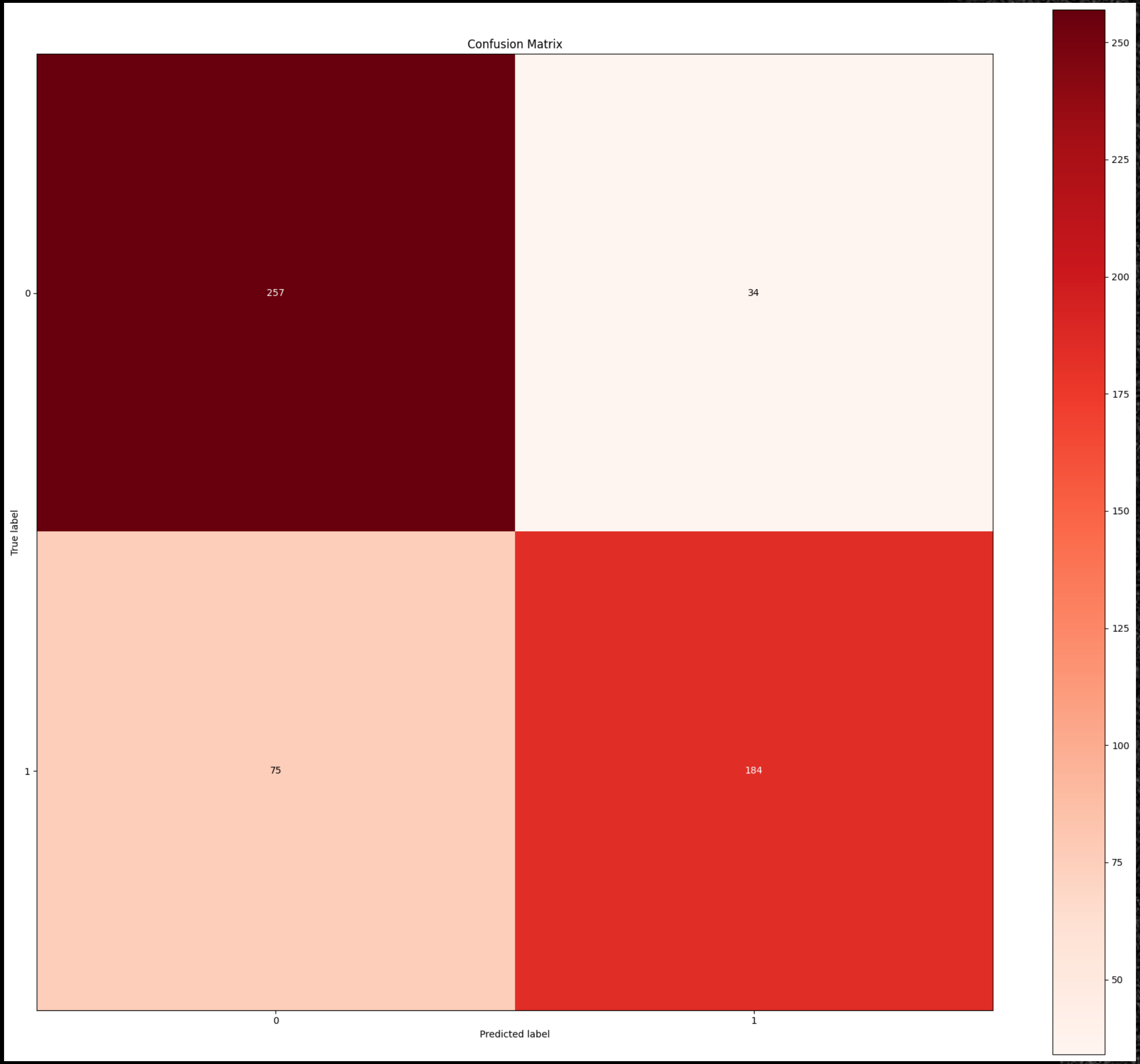
LMST Cross validation



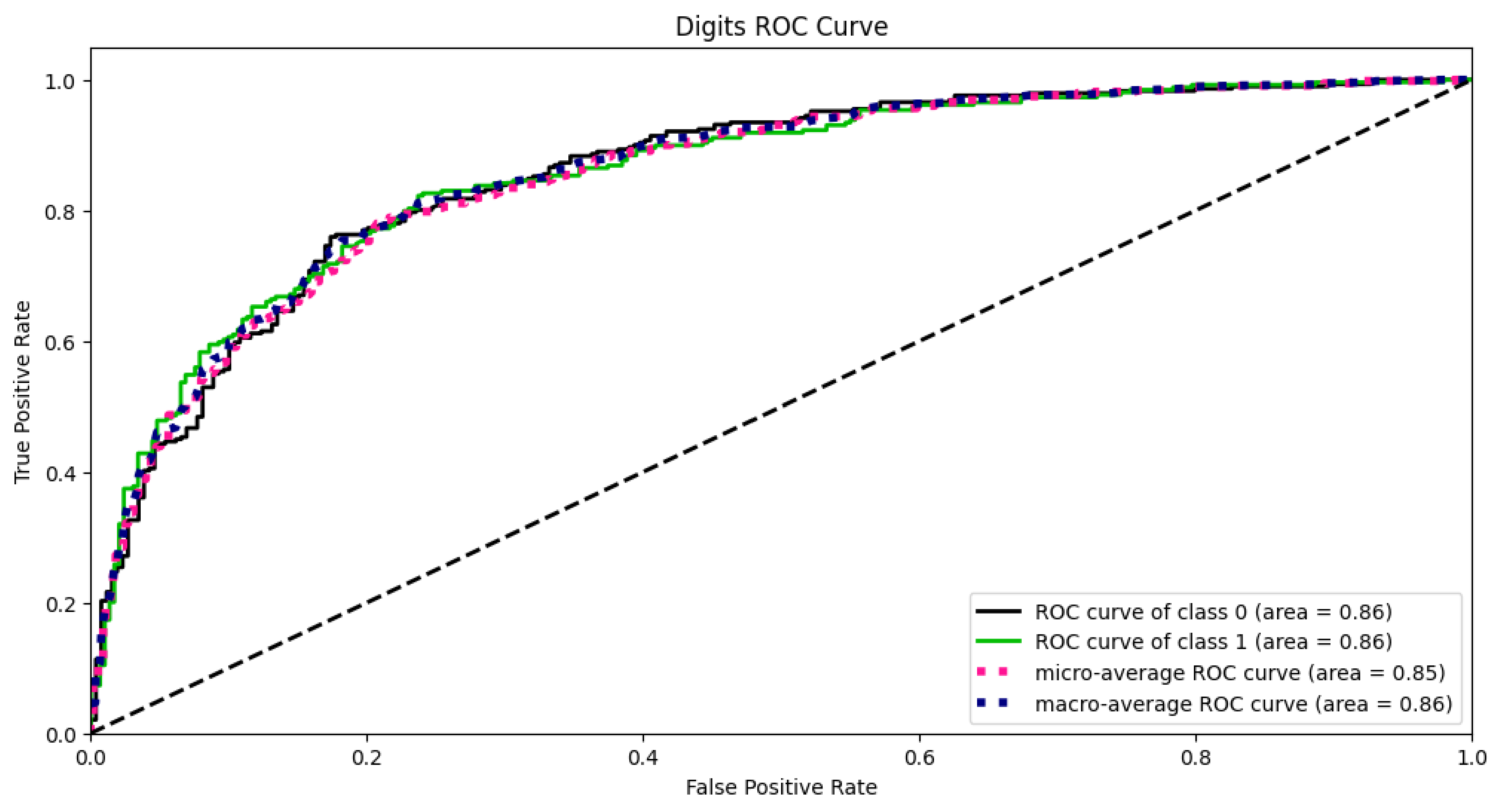
Train Score vs Cross Validation



RNN Cross validation


Confusion Matrix
True label
Predicted label
0
1
0
1
257
34
75
184
250
225
200
175
150
125
100
75
50

ROC curve


Digits ROC Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve of class 0 (area = 0.86)
ROC curve of class 1 (area = 0.86)
micro-average ROC curve (area = 0.85)
macro-average ROC curve (area = 0.86)

Conclusions

- Conclusions:
- The Dummy Classifier provides a baseline, and any model that cannot outperform its metrics is ineffective.
- The RNN shows significant improvement over the Dummy Classifier, but the LSTM outperforms both in terms of accuracy and overall performance.
- Accuracy and recall are especially high for the LSTM, indicating that it can correctly classify the two classes effectively.

Opportunities for Improvement:

- Explore Other Models
- Hyperparameter Optimization
- Feature Engineering
- Deployment