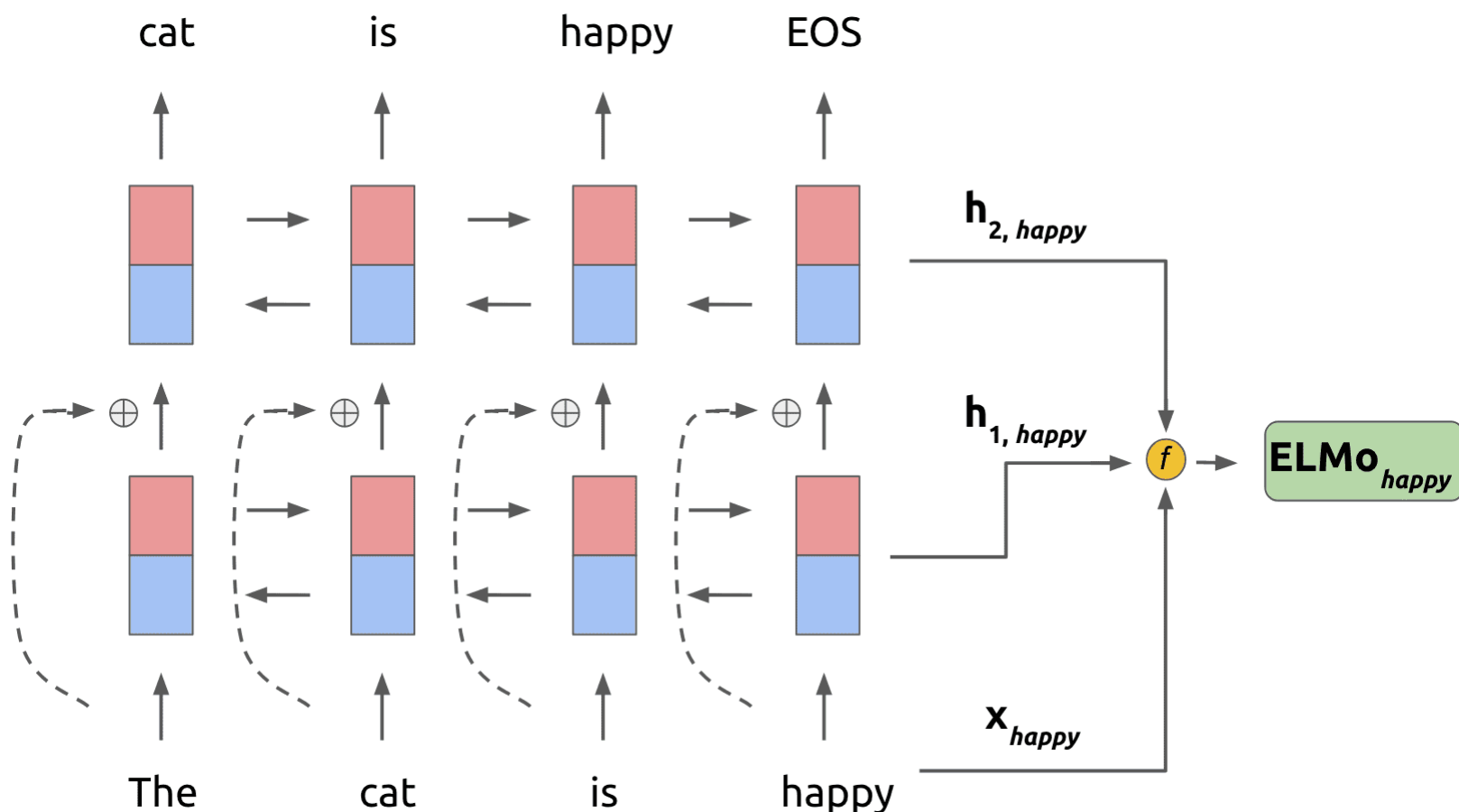


Assignment 4

Report

ELMo Architecture



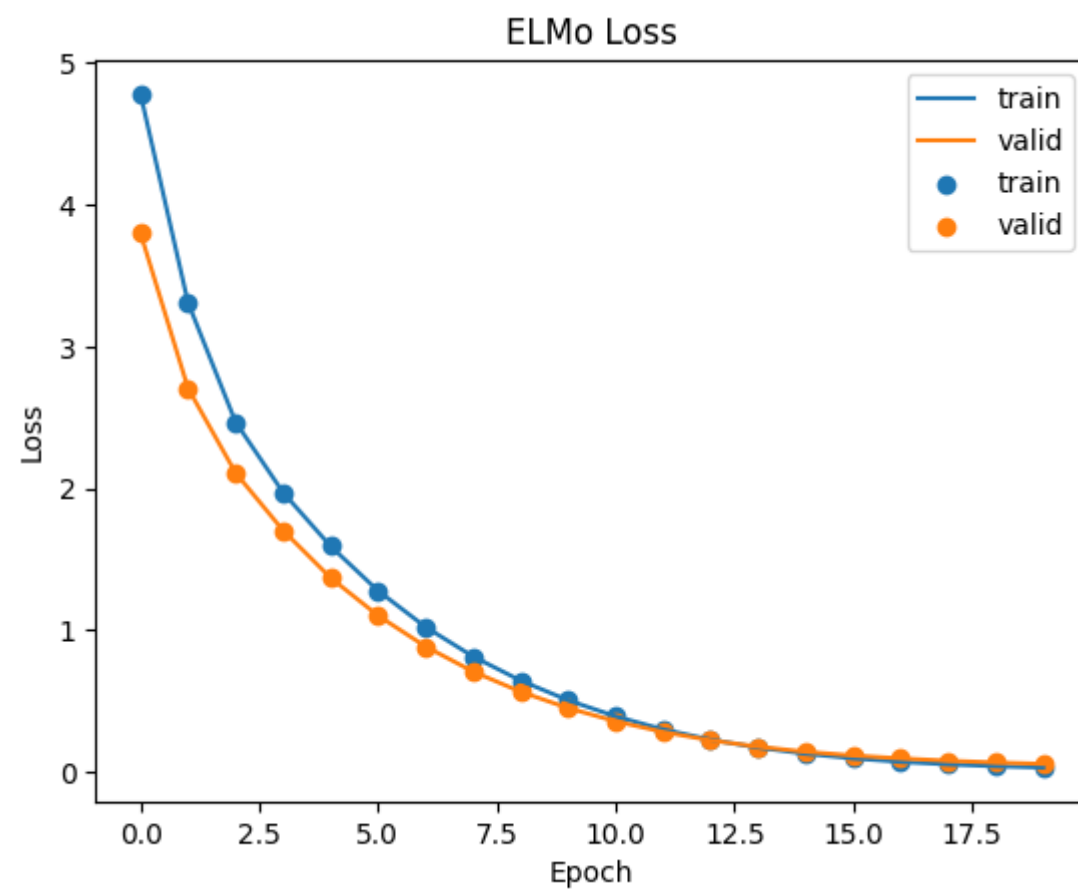
- The model takes in an input tensor of size $(batch_size, sequence_length)$ where each element in the tensor is a word index representing a token in the input text.
- The `embedding` layer looks up the pretrained GloVe embeddings for each word in the input tensor and returns a tensor of size $(batch_size, sequence_length, embedding_dim)$ where `embedding_dim` is the size of the GloVe embedding vectors.
- The `lstm1` layer is a bidirectional LSTM layer with `hidden_dim` hidden units, which means it has `hidden_dim` hidden units in both the forward and backward directions. This layer takes in the output of the `embedding` layer and returns a tensor of size $(batch_size, sequence_length, hidden_dim*2)$ because the outputs of the forward and backward LSTMs are concatenated along the last dimension.
- The `lstm2` layer is also a bidirectional LSTM layer with `hidden_dim` hidden units that takes in the output of `lstm1` and returns a tensor of size $(batch_size, sequence_length, hidden_dim*2)$.
- The `linear_out` layer is a linear layer that takes in the output of `lstm2` and returns a tensor of size $(batch_size, sequence_length, vocab_size)$ where `vocab_size` is the number of output classes (i.e. the size of the output vocabulary).

Overall, the model is designed to take in a sequence of word indices, embed those words using pretrained GloVe embeddings, and then use a bidirectional LSTM to encode the sequence into a fixed-length vector representation that can be used for classification or other downstream tasks.

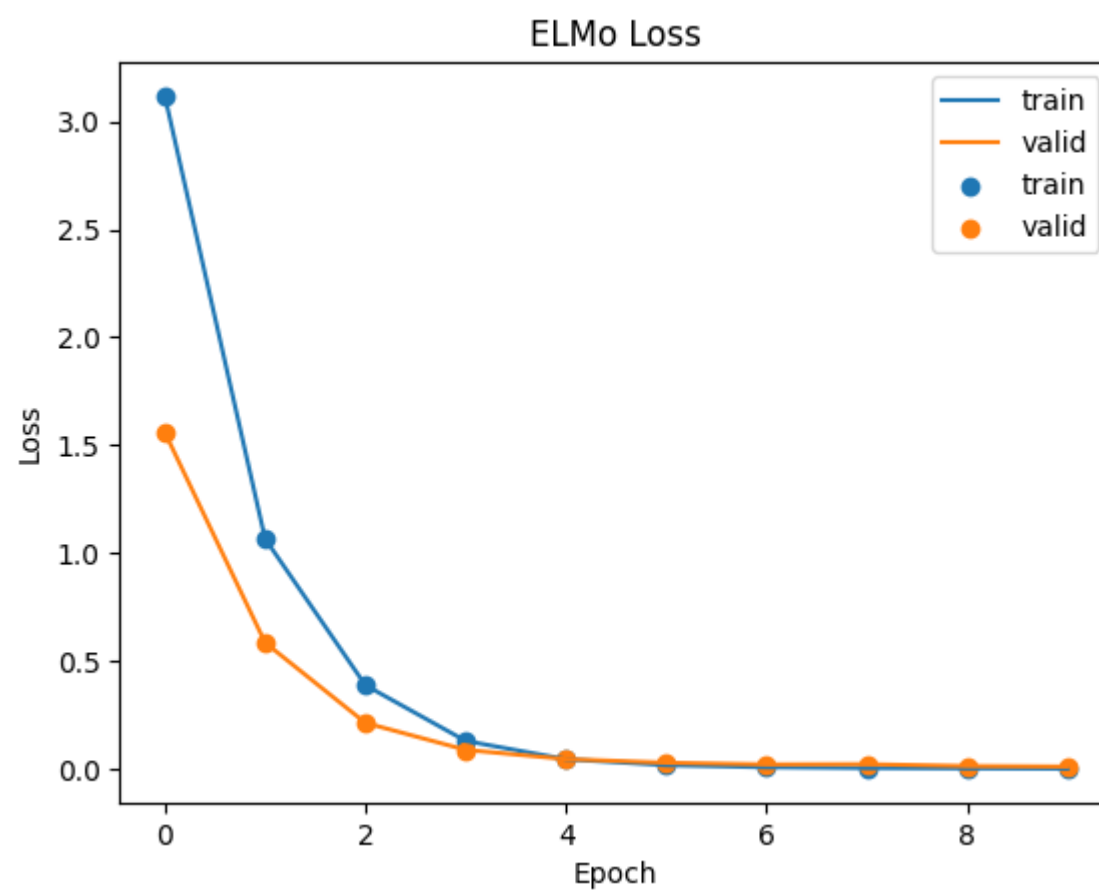
```
ELMo(  
  (embedding): Embedding(10942, 300, padding_idx=0)  // Glove pretrained embedding  
  (lstm1): LSTM(300, 100, batch_first=True, bidirectional=True)  
  (lstm2): LSTM(200, 100, batch_first=True, bidirectional=True)  
  (linear1): Linear(in_features=300, out_features=100, bias=True)  
  (linear_out): Linear(in_features=200, out_features=10942, bias=True)  
)
```

The inputs are the the sequence of word indices from 0 to length-1 , and the output is the sequence of word indices from 1 to length .

Dataset : Stanford Sentiment Treebank



Dataset : Multi-Genre NLP corpus



Classification

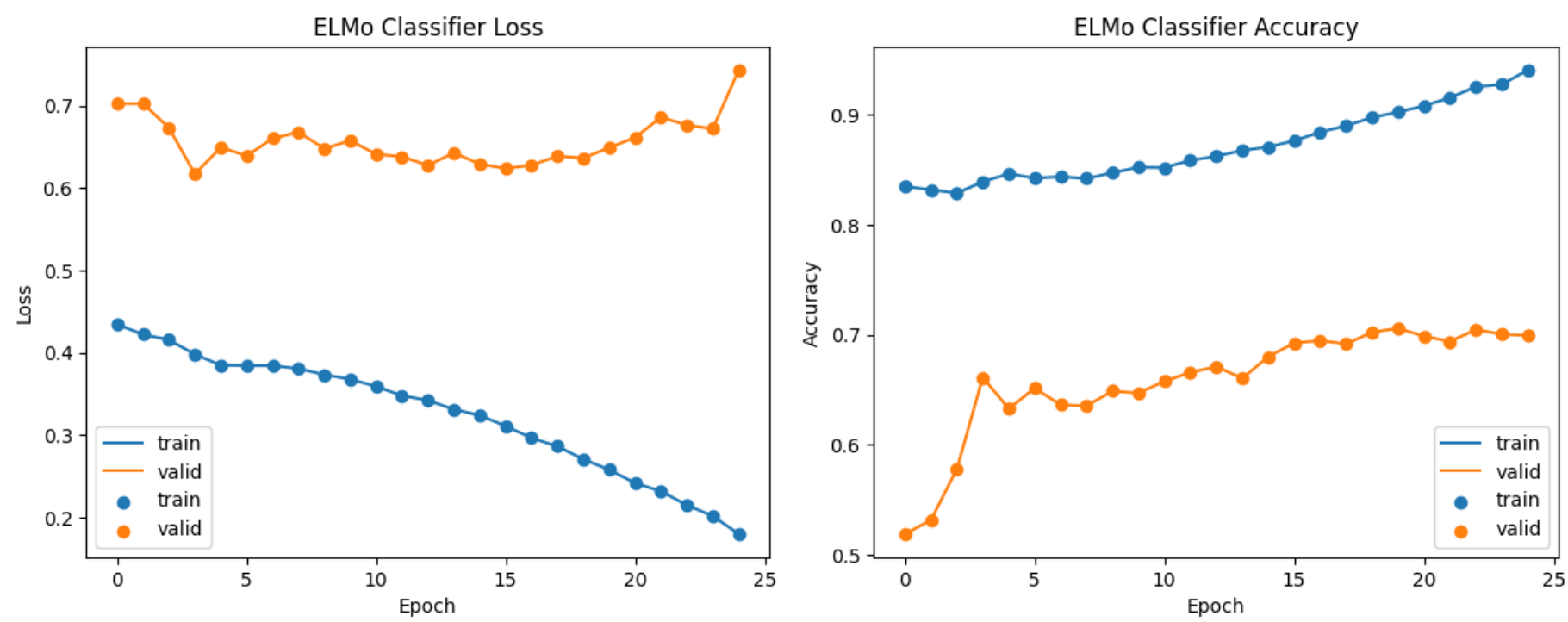
The architecture consists of the following layers:

- `nn.Embedding`: The input is first passed through an embedding layer using the pretrained ELMo embedding.
- `nn.Linear`: The output from the embedding layer is passed through a linear layer to change the embedding dimension to $\text{hidden dimension} \times 2$.
- `nn.LSTM`: The resulting tensor is then passed through two layers of LSTM in a sequential manner.
- `torch.max`: After passing through the LSTM layers, the resulting tensor is passed through a max pooling layer.
- `nn.Dropout`: The max pooling output is then passed through a dropout layer with a dropout probability of 0.5.
- `nn.Linear`: Finally, the resulting tensor is passed through a linear layer to get the output of the classifier.

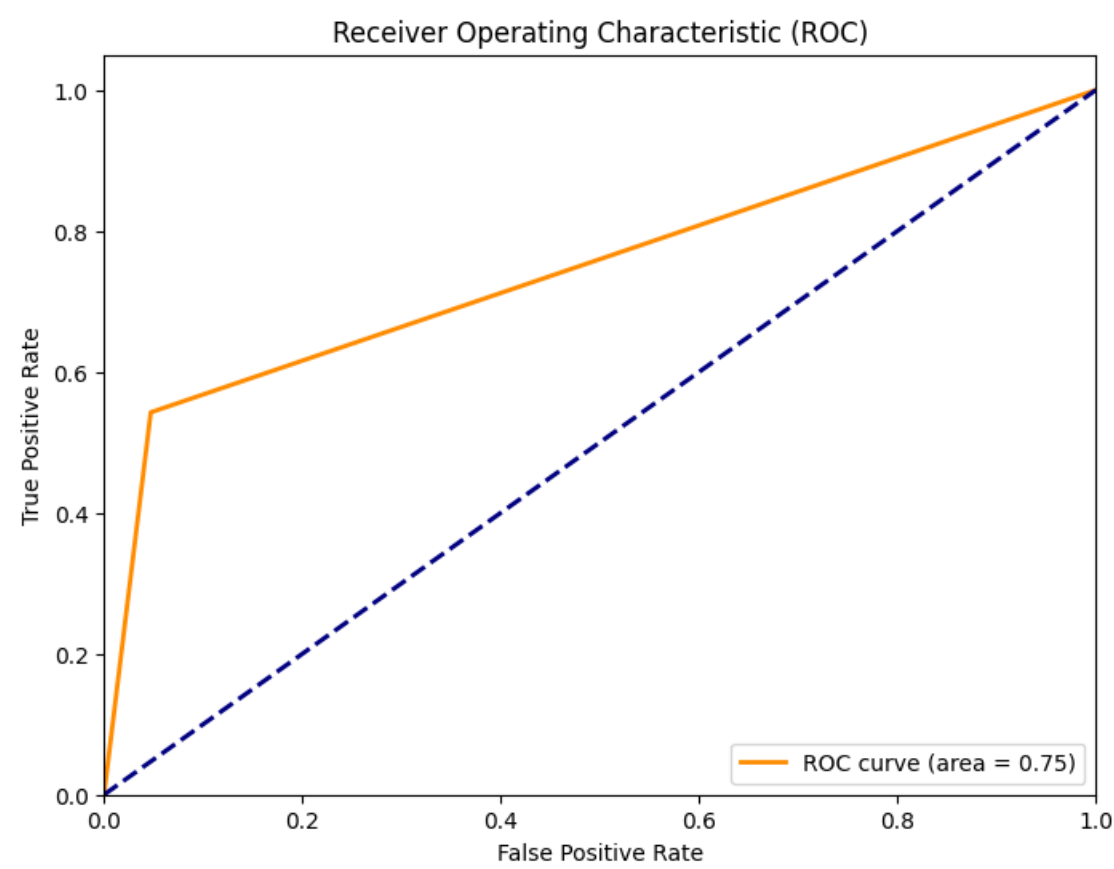
This architecture also includes L2 regularization.

```
scoreClassifier(
    (embeddings): Embedding(10942, 300, padding_idx=0)
    (lstm1): LSTM(300, 100, batch_first=True, bidirectional=True)
    (lstm2): LSTM(200, 100, batch_first=True, bidirectional=True)
    (linear1): Linear(in_features=300, out_features=200, bias=True)
    (linear_out): Linear(in_features=200, out_features=3, bias=True)
    (dropout): Dropout(p=0.5, inplace=False)
)
```

Dataset : Stanford Sentiment Treebank



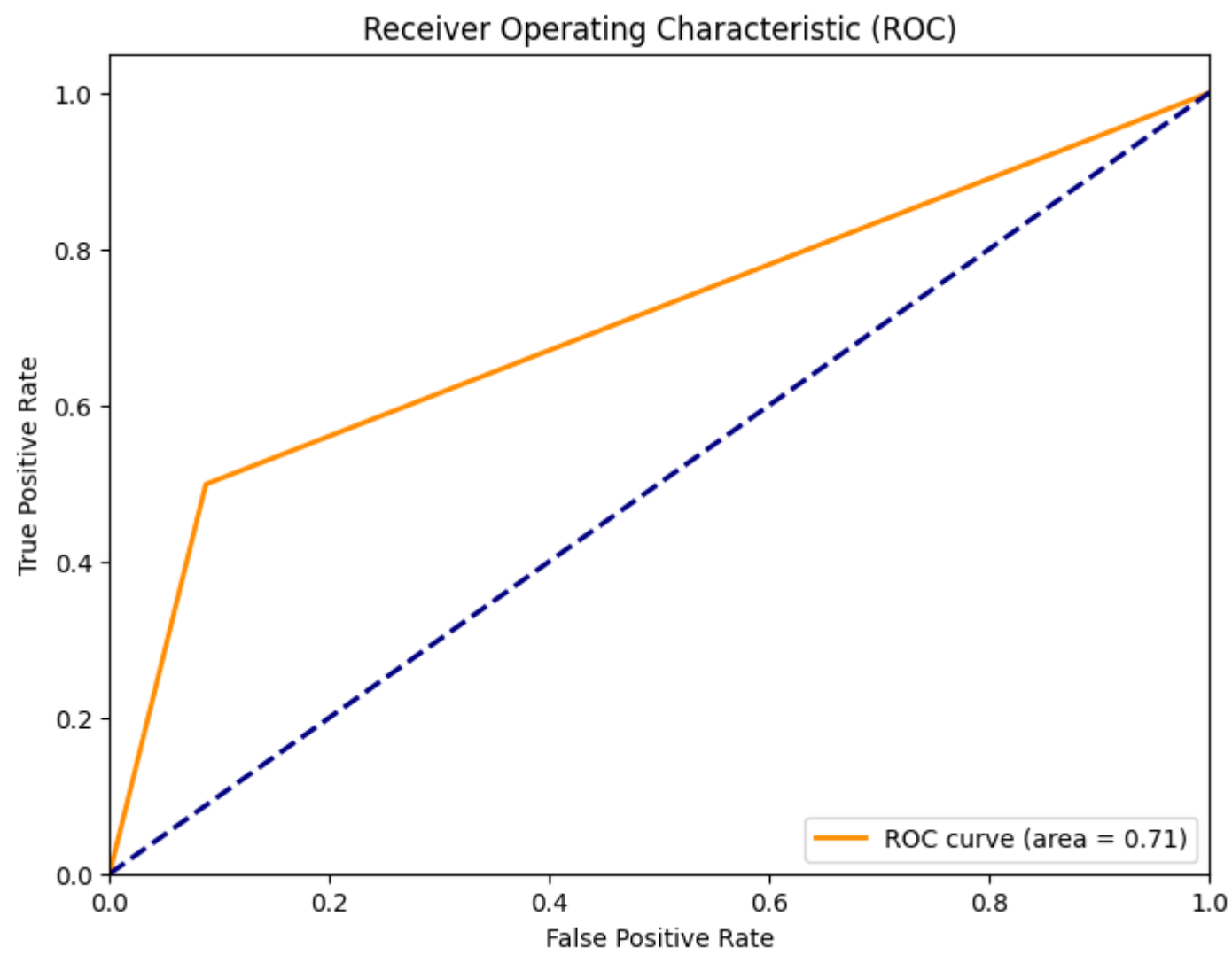
Visualizations



Training Dataset					
	precision	recall	f1-score	support	
	0	0.67	0.95	0.79	4244
	1	0.92	0.54	0.68	4300
accuracy				0.75	8544
macro avg	0.80	0.75	0.74		8544
weighted avg	0.80	0.75	0.74		8544

Accuracy: 0.7463717228464419
F1 Score: 0.6828625786623739
Precision: 0.9210422424003158
Recall: 0.5425581395348837
Confusion Matrix:
[[4044. 200.]
 [1967. 2333.]]

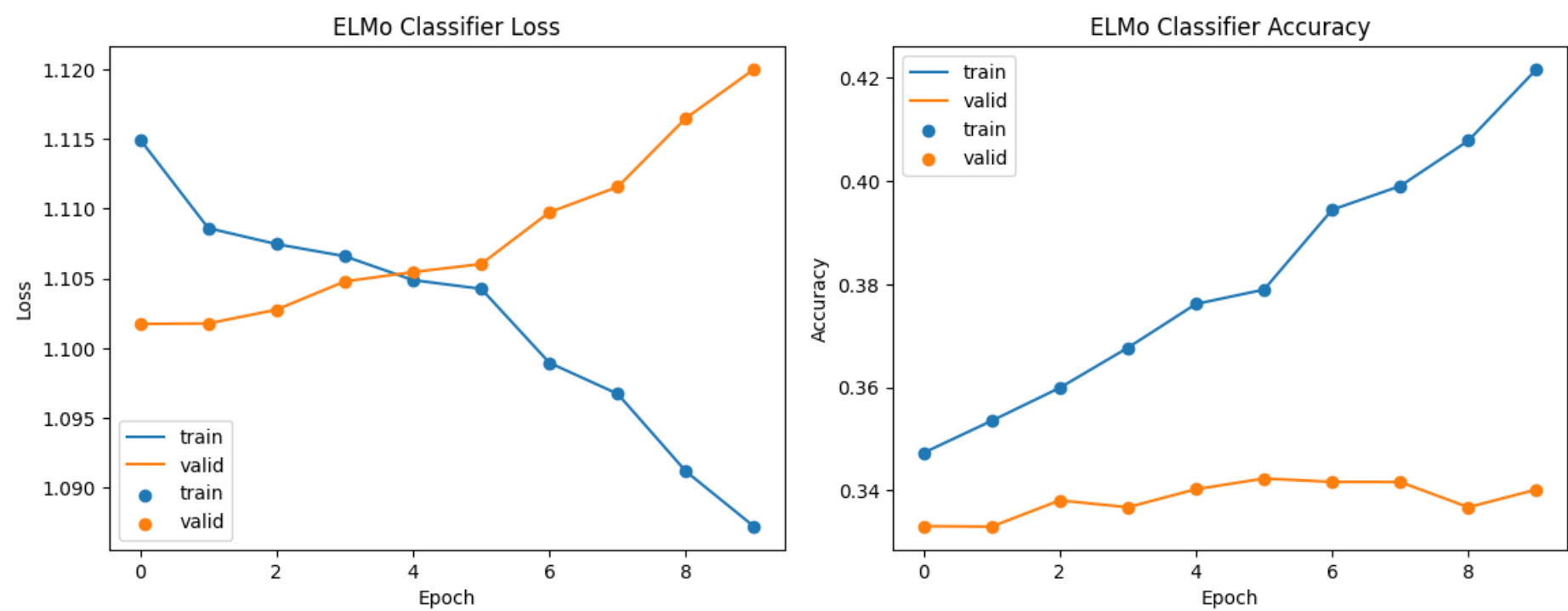
Test Dataset



	precision	recall	f1-score	support
0	0.66	0.91	0.77	1143
1	0.84	0.50	0.63	1067
accuracy			0.71	2210
macro avg	0.75	0.71	0.70	2210
weighted avg	0.75	0.71	0.70	2210

Accuracy: 0.7126696832579186
F1 Score: 0.6262507357268982
Precision: 0.8417721518987342
Recall: 0.4985941893158388
Confusion Matrix:
[[1043. 100.]
 [535. 532.]]

Dataset: Multi-Genre NLP corpus



Visualizations

Training Dataset					
	precision	recall	f1-score	support	
negative		0.37	0.21	0.27	10371
neutral		1.00	0.00	0.00	9130
positive		0.37	0.82	0.51	11105
accuracy				0.37	30606
macro avg		0.58	0.34	0.26	30606
weighted avg		0.56	0.37	0.27	30606
Accuracy: 0.3679997386133438					
Confusion Matrix:					
[[2153. 0. 8218.]					
[1645. 0. 7485.]					
[1995. 0. 9110.]]					
Testing Dataset					
	precision	recall	f1-score	support	
negative		0.35	0.14	0.20	3463
neutral		1.00	0.00	0.00	3129
positive		0.33	0.86	0.48	3240
accuracy				0.33	9832

macro avg	0.56	0.33	0.23	9832
weighted avg	0.55	0.33	0.23	9832

Accuracy: 0.33329943043124494

Confusion Matrix:

```
[[ 492.    0. 2971.]
 [ 466.    0. 2663.]
 [ 455.    0. 2785.]]
```