

```

42 )
43 return model

```

Code Listing 5: Defining loss function ; projection head and the classifier [10]

- The classifier is trained with the frozen encoder using the text and image inputs. The classifier has one layer using the Relu function and another Softmax layer and is also trained for 10 epochs. The following code shows the classifier with the frozen encoder using 2 inputs.

```

1
2 #Pretrain the encoder
3
4 encoder = create_encoder()
5
6 encoder_with_projection_head = add_projection_head(encoder)
7 encoder_with_projection_head.compile(
8     optimizer=keras.optimizers.Adamax(learning_rate),
9     loss=SupervisedContrastiveLoss(temperature),
10 )
11 history = encoder_with_projection_head.fit([X_train, trainingpadded],
12     Y_train, batch_size=batch_size, epochs=num_epochs)
13
14 # Train the classifier with the frozen encoder
15
16 classifier = create_classifier(encoder, trainable=False)
17
18 history = classifier.fit([X_train, trainingpadded], Y_train, batch_size=
19     batch_size, epochs=num_epochs, validation_data=([X_test, testingpadded],
20     Y_test))
21 accuracy = classifier.evaluate([X_test, testingpadded], Y_test)[1]
22
23 print(f"Test accuracy: {round(accuracy * 100, 2)}%")

```

Code Listing 6: Training the classifier with the frozen encoder [10]

5 Empirical Evaluation

Accuracy represents the number of the correct predicted labels divided by the total number of the samples that were classified [3]. The results below compare the baseline