

CNN-1D Assignment

Yuhuan Huang

Console output:

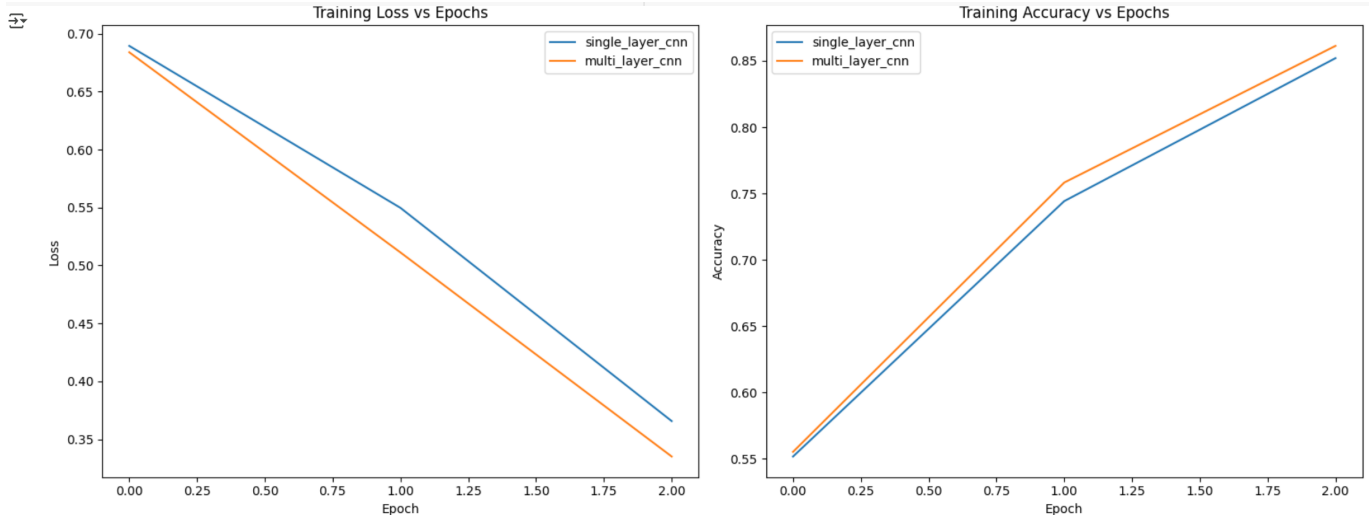
```
>>> np.isclose(models['multi_layer_cnn'].evaluate(X_test, y_test, verbose=0)[0], 0.4696541130542755, atol=1e-1)
np.True_
>>> np.isclose(models['multi_layer_cnn'].evaluate(X_test, y_test, verbose=0)[1], 0.796999990940094, atol=1e-1)
np.True_
"""
doctest.testmod()
```

```
Random seeds have been set to: 42
Random seeds have been set to: 42
Loading IMDB dataset...
Dataset loaded. Training samples: 25000, Test samples: 25000
Using 5000 training samples and 1000 test samples

Creating models...
Random seeds have been set to: 42
Random seeds have been set to: 42

Training single_layer_cnn
Random seeds have been set to: 42
Random seeds have been set to: 42
Epoch 1/3
/usr/local/lib/python3.12/dist-packages/keras/src/layers/core/embedding.py:97: UserWarning: Argument `input_length` is deprecated. Just remove it.
  warnings.warn(
125/125 ————— 5s 16ms/step - accuracy: 0.5243 - loss: 0.6930 - val_accuracy: 0.6890 - val_loss: 0.6607
Epoch 2/3
125/125 ————— 2s 15ms/step - accuracy: 0.7112 - loss: 0.5998 - val_accuracy: 0.7770 - val_loss: 0.4749
Epoch 3/3
125/125 ————— 2s 15ms/step - accuracy: 0.8345 - loss: 0.4063 - val_accuracy: 0.8140 - val_loss: 0.4239

Training multi_layer_cnn
Random seeds have been set to: 42
Random seeds have been set to: 42
Epoch 1/3
125/125 ————— 5s 27ms/step - accuracy: 0.5234 - loss: 0.6923 - val_accuracy: 0.7090 - val_loss: 0.6026
Epoch 2/3
125/125 ————— 5s 23ms/step - accuracy: 0.7436 - loss: 0.5447 - val_accuracy: 0.7770 - val_loss: 0.4784
Epoch 3/3
125/125 ————— 5s 23ms/step - accuracy: 0.8519 - loss: 0.3596 - val_accuracy: 0.7580 - val_loss: 0.5355
```



```

Final Training Metrics:

single_layer_cnn:
  Loss: 0.3657
  Accuracy: 0.8520
  Test Accuracy: 0.8160

multi_layer_cnn:
  Loss: 0.3351
  Accuracy: 0.8612
  Test Accuracy: 0.7700
Random seeds have been set to: 42
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/imdb\_word\_index.json
1641221/1641221 — 0s 0us/step

Example 1:
Review (truncated): graphics not up to 2003 standards br br the action is presented as a series of flashbacks the aged is so we get a quick look at some historical events some of which are presented we
True sentiment: Negative
1/1 — 0s 190ms/step
single_layer_cnn prediction: Negative (confidence: 0.93)
1/1 — 0s 169ms/step
multi_layer_cnn prediction: Negative (confidence: 0.84)

Example 2:
Review (truncated): spoilers br br where do i begin with just how silly...
True sentiment: Negative
1/1 — 0s 62ms/step
single_layer_cnn prediction: Negative (confidence: 0.96)
1/1 — 0s 64ms/step
multi_layer_cnn prediction: Negative (confidence: 0.96)

Example 3:
Review (truncated): ...
True sentiment: Negative
1/1 — 0s 61ms/step
single_layer_cnn prediction: Negative (confidence: 0.95)
1/1 — 0s 68ms/step
multi_layer_cnn prediction: Negative (confidence: 0.95)

Example 4:
Review (truncated): this soft soft core sci fi b movie is what you'd have if you took an early fred ray film and took out the fun or it's like an uwe boll but without as much a young chain gang co
True sentiment: Negative
1/1 — 0s 85ms/step
single_layer_cnn prediction: Negative (confidence: 0.52)
1/1 — 0s 68ms/step
multi_layer_cnn prediction: Negative (confidence: 0.81)

Example 5:
Review (truncated): in a mass of real stupid ones if one expects all dialogues to be absurd the fun wears off br br you see there is american pie 2 my all time favorite teenager movie it contains
True sentiment: Negative
1/1 — 0s 77ms/step
single_layer_cnn prediction: Negative (confidence: 0.69)
1/1 — 0s 73ms/step
multi_layer_cnn prediction: Positive (confidence: 0.81)
TestResults(failed=0, attempted=8)

```

Reflections:

Review your notebook results and answer the following reflection questions:

1. How did the single-layer CNN perform compared to the multi-layer CNN?

My Ans: From the two graphs, the change in loss of single-layer CNN resembles that of multi-layer CNN. However, we can see that multi-layer CNN outperforms the single-layer CNN, which makes sense since single-layer CNN is a relatively "simpler" model.

2. What are the advantages of using CNNs for text classification?

My Ans: Firstly, we don't need much text preprocessing, CNN can automatically extract features, so it would be "easier to use" than the other text classification models. Also, CNN is good at detecting n-gram features and capture patterns.

3. How might you further improve these models?

My Ans: I might improve the model by tuning for better hyperparameters and adjusting the pipelines. Also, I can add the embedding methods, which we learned from the last two classes to this CNN.

4. Why might the single-layer CNN have outperformed the multi-layer CNN in our experiments?

My Ans: We only have 3 epochs, and this may not be enough for a complex model. Also, maybe we can find better hyperparameters for the multi-layer CNN model.