

Question 3.1.1 Review the data to see how it is organized in the file. How many examples are in the file data.tsv?

Data contains two columns. The text column contains a string(fragment of the sentence or single/multiple completed sentences), the label column stores value {0,1}: 0 represents objective and 1 represents subjective.

There are in total 10000 examples in the file.

Question 3.1.2 For all four examples, explain, in English, why it has the given label.

Subjective examples:

“it's mildly amusing, but i certainly can't recommend it.”

This example expresses personal feeling and provide a point of view of audience's future action.

“the film's sense of imagery gives it a terrible strength, but it's propelled by the acting.”

This example criticizes the film's sense of imagery and approves acting. It contains personal judgement towards the film.

Objective examples:

“a young man is found bruised, beaten and stumbling down a secluded road.”

“in 1959, berry gordy jr. gathered the best musicians from detroit's thriving jazz and blues scene to begin cutting songs for his new record company.”

Both examples describe a scene or an experience recorded in the movie, which does not involve personal feeling/judgements.

Question 3.1.3 Find one example from each of the positive and negative sets that you think has the incorrect label and explain why each is wrong.

“a selection of scenes in search of a movie”

This example is labeled as subjective in the dataset. I do not observe any personal opinion/feelings/judgement in this sentence, instead, this rather describe the fact that scenes are selected when finding the movie.

“unfortunately the third seems out of reach - the beautiful, seductive anne elise.”

This example is labeled as objective in the dataset, however, “unfortunately”, “beautiful”, and “seductive” words indicate an opinion in the sentence. “the third seems out of reach” is considered to be negative, and the sentence views “anne elise” as beautiful and seductive.

Question 3.2 You should verify that there are equal number of examples in the two classes in each of the datasets, and that you did not accidentally put the same sample in more than one of the sets.

```
Checking label distribution in train dataset
label
0    3200
1    3200
Name: count, dtype: int64
Checking label distribution in validation dataset
label
0     800
1     800
Name: count, dtype: int64
Checking label distribution in test dataset
label
1    1000
0    1000
Name: count, dtype: int64
Checking label distribution in overfit dataset
label
0     25
1     25
Name: count, dtype: int64
If we concatenate train, validation, and test, drop duplicates(if applies...)
the length is: 10000
```

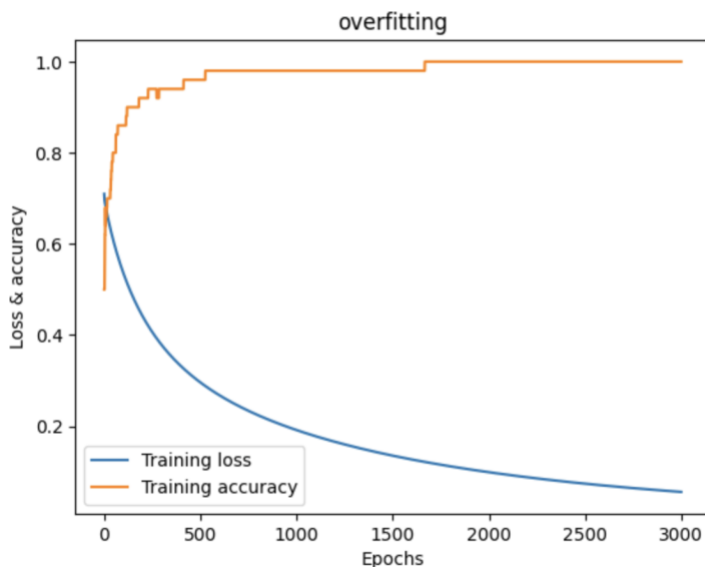
Use `value_counts` and verified that there are equal number of label 0 and label 1 in all datasets. If we concatenate all three datasets, drop duplicates, we found the length is equal to the original dataset. This proves there are no duplicated rows used across different sets.

Question 4.4

Provide the training loss and accuracy plot for the overfit data in your Report.

Train the model in 3000 epochs, the model can achieve training accuracy = 1.0, while validation accuracy = 0.84.

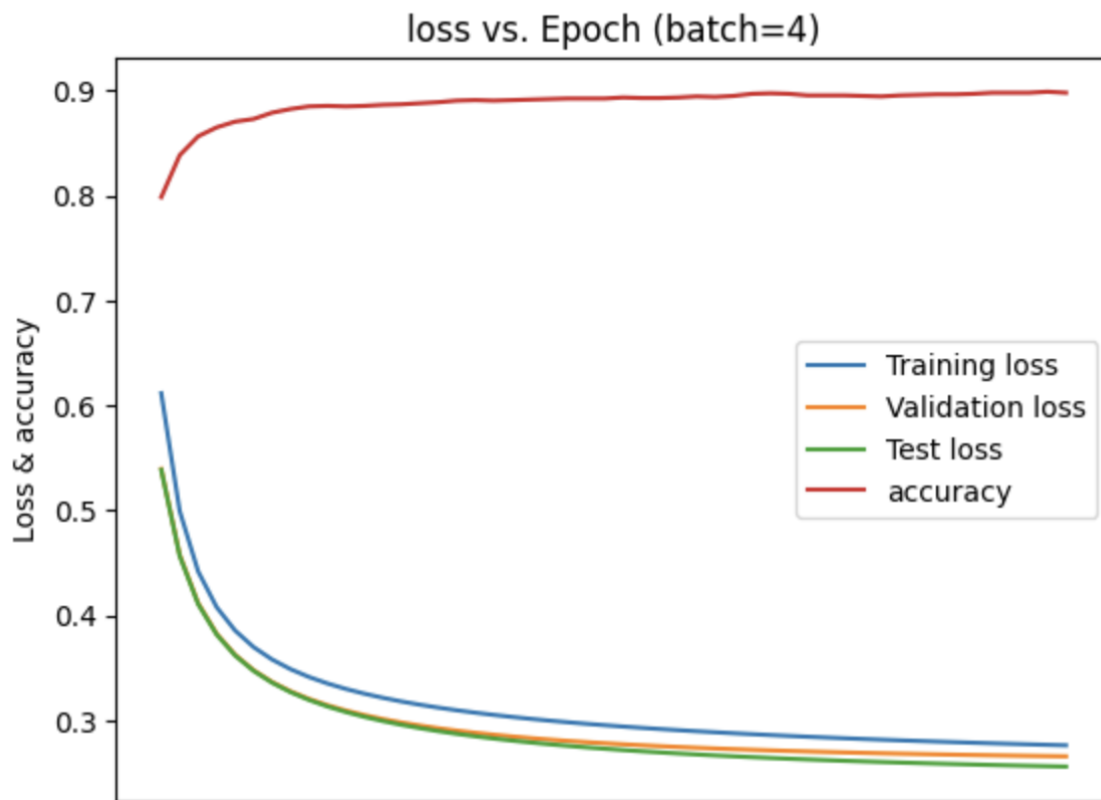
```
Using device: cpu
final train accuracy: 1.0
final validation accuracy: 0.83875
```



Question 4.5

Given the training and validation loss and accuracy curves vs. epoch in your report, and report the final test accuracy. Evaluate the test data and provide the accuracy result in your report.

Epoch 50, Validation Loss: 0.26573233065428215
Epoch 50, Test Loss: 0.2559765983284451
Epoch 50, Accuracy: 0.898



The final test accuracy = 0.898. All loss curves went down smoothly during training, validation and test loss does not go up, indicating no overfitting behaviors.

In the baseline model, what information contained in the original sentence is being ignored?

In this baseline model, we are ignoring any word that could not be properly mapped to a word vectors using glove, and any punctuations in the sentence.

Question 4.6

Determine the 20 closest words to those trained parameters of the neuron. Do some of the words that you generated make sense? Explain.

Most of the words are describing the subjective feelings, such as “pleasing”, “underwhelming”, “funnily”, “overacting”. Therefore, it seems like the baseline model first average the meaning of the entire sentence into one word embedding, and try to compare this average embedding with the neuron parameters(a “word embedding” that leans toward subjective).

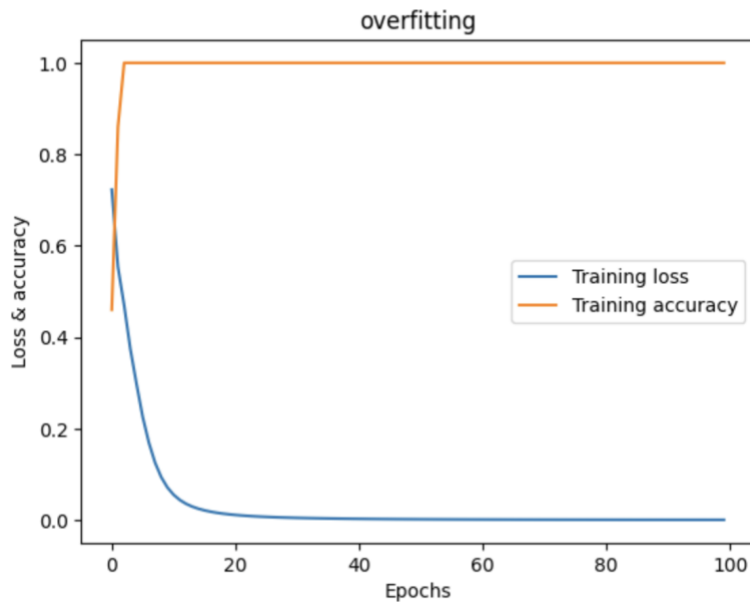
```
print_closest_cosine_words(model_4.linear.weight[0].detach(), n=20)💡
```

```
⇒ insipid      0.53
   pleasing    0.51
   verbiage    0.51
   emptier     0.50
   underwhelming 0.50
   campy       0.50
   simplistic   0.50
   funnily     0.50
   watchable   0.49
   appetizing  0.49
   nuanced     0.49
   laudatory   0.49
   tepid       0.48
   visuals     0.48
   succinct    0.48
   laughable   0.48
   uninspired  0.47
   crispness   0.47
   glib        0.47
   overacting  0.47
```

Question 5.1

I am able to achieve training accuracy = 1.0, while validation accuracy = 0.865.

```
Epoch 100, Train Accuracy: 1.0
final train accuracy: 1.0
final validation accuracy: 0.865
```



Question 5.2.1

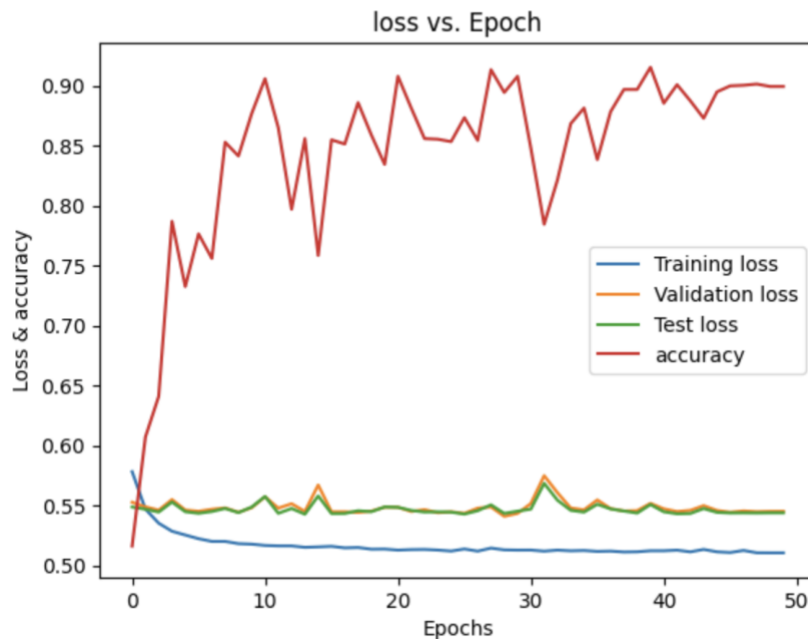
Report the accuracy and the full hyperparameter settings. Give the training and validation curves for that best model, and describe your overall hyperparameter tuning approach.

Epoch = 50, batch size = 4, learning rate = 0.001, k1 = 2, n1 = 50, k2 = 4, n2 = 80

The final accuracy is 0.8995.

First, I found that epoch = 50, batch size = 4, and learning rate = 0.001 guarantees a better result. Then I started to experiment with k1 and k2, I found a k1-k2 pair in <2,3>, <2,4>, <3,4> could achieve a similar level of good result. So I decide to stick with k1-k2 pair <2,4>. At the end, I tunned n1, n2 and found n1 = 50, n2 = 80 provided the best result in terms of accuracy.

```
Epoch 50, Train Loss: 0.5103832307830453
Epoch 50, Validation Loss: 0.5450485632568598
Epoch 50, Test Loss: 0.5437365266680717
Epoch 50, Accuracy: 0.8995
```



Question 5.2.2

Report the accuracy of the result, and comment on the result.

The final accuracy is 0.898, which is a slight decrease compared to the model result when freeze is set (accuracy is 0.8995).

It seems that the pre-trained embeddings are ready for the task, so tuning the initial word embeddings during training does not bring any benefits but some noises that drag the accuracy a little bit. Additionally, it is time-consuming.

Since our dataset is quite small, it may be a better idea to freeze the word embeddings instead of tuning them while training.

Question 5.3

Use that function to determine the five closest words to each of the words in the kernels trained in your best classifier. Do those words make sense? Do the set of words in each given kernel give a broader insight into what the model is looking for? Explain.

My conv1 layers has k1 = 2, I chose to average across this 2 vectors to get(1,100), since there are n1 kernels, I decided to checked the first 3 kernels. Similar approaches were applied to conv2 layer. For each kernel(3 from conv1 and 3 from conv2), I got the 5 closest words shown below.

Most of the words in the set are nouns or verbs, which could be generally used when describing a scene in the movie. Therefore, it could be concluded that the model is looking for those

descriptive words to classify the sentence into objective, otherwise it will lean more to classify it as subjective.

```
✓ ---the 1th kernel in conv1 has following 5 closest words:---
tavernier      0.52
writer-director 0.50
imax           0.50
auteur         0.48
monodrama      0.48
---the 1th kernel in conv2 has following 5 closest words:---
befriended     0.49
17-year-old    0.49
kung-fu        0.48
recruited      0.47
kidnap         0.47
---the 2th kernel in conv1 has following 5 closest words:---
raided         0.40
according      0.39
temple         0.38
phalangist     0.38
centrebus      0.38
---the 2th kernel in conv2 has following 5 closest words:---
quantel        0.59
sextuplet      0.58
ayisha         0.57
66-acre        0.57
fin.k.l        0.56
---the 3th kernel in conv1 has following 5 closest words:---
obsessed       0.52
kung-fu        0.52
ex-husband     0.52
free-spirited  0.52
prostitute     0.50
---the 3th kernel in conv2 has following 5 closest words:---
teenager       0.43
hamilton       0.42
student        0.42
activist       0.40
recruited      0.40
```

Question 6.1

Include the input and output in our write up. Comment on how the two models performed and whether they are behaving as you expected. Do they agree with each other? Which model seems to be performing the best?

Below are the two sentences that are definitely objective and subjective.

<p>sentence</p> <p>This is written in the book</p> <p>Clear Submit</p>	<p>output</p> <p>baseline determines it as Objective with probability 0.018485713750123978 cnn determines it as Objective with probability 1.3430671685910056e-08</p> <p>Flag</p>
<p>sentence</p> <p>I am so happy</p> <p>Clear Submit</p>	<p>output</p> <p>baseline determines it as Subjective with probability 0.9999020099639893 cnn determines it as Subjective with probability 0.9998891353607178</p> <p>Flag</p>

Below are the two sentences that are borderline objective and subjective.

<div>sentence</div> <div>Surprisingly this was his last class in the university</div> <div>Clear Submit</div>	<div>output</div> <div>baseline determines it as Objective with probability 0.055497512221336365 cnn determines it as Objective with probability 8.970105214167745e-11</div> <div>Flag</div>
<div>sentence</div> <div>This is the scene that described in that movie</div> <div>Clear Submit</div>	<div>output</div> <div>baseline determines it as Subjective with probability 0.9990819692611694 cnn determines it as Objective with probability 0.01634126901626587</div> <div>Flag</div>

Two models agree with each other when the sentence is definitely objective/subjective. But it may predict different results when the sentence is borderline subjective/objective. CNN seems to be performing better than the baseline due to its complex layer structures.

“This is the scene that described in that movie”: cnn managed to classify this as objective.

“Surprisingly this was his last class in the university”: although both baseline and cnn classify this as objective, we can observe that baseline generates a higher result (more lean to 1). It may be misled by the word “surprisingly”.