Assignment 2

Hruday Vishal Kanna Anand

S.No- 1006874517

Q3.1.1 Review the data to see how it is organized in the file. How many examples are in the file data.tsv?

The Data is organized as a tsv file with 2 columns the first column contains the sentence or phrase and the second column contains the label which is either 0 or 1. These 2 columns are separated by a tab.

There are a total of 10,000 examples-

	text	label
0	smart and alert , thirteen conversations about...	1
1	color , musical bounce and warm seas lapping o...	1
2	it is not a mass-market entertainment but an u...	1
3	a light-hearted french film about the spiritua...	1
4	my wife is an actress has its moments in looki...	1
...	...	...
9995	in the end , they discover that balance in lif...	0
9996	a counterfeit 1000 tomin bank note is passed i...	0
9997	enter the beautiful and mysterious secret agen...	0
9998	after listening to a missionary from china spe...	0
9999	looking for a short cut to fame , glass concoc...	0

10000 rows × 2 columns

Q3.1.2 Select two random examples each from the positive set (subjective) and two from the negative set. For all four examples, explain, in English, why it has the given label.

Objective examples-

```
get_random_data(df)
```

```
("this film is adapted from the play the fire and the rain by one of india's foremost playwrights , girish karnad . ",
 0)
```

```
get_random_data(df)
```

```
("before he can do anything about it , though , jed is seduced by jackson's lesbian artist friend . ",
 0)
```

From both the examples above it can be seen that these sentences are explaining something that is happening in the film and therefore can be considered as facts. Hence we can say that they are objective

Subjective examples

```
get_random_data(df)
```

```
('those who want to be jolted out of their gourd should drop everything and run to ichi . ',
 1)
```

```
get_random_data(df)
```

```
("it's shocking , intense , and totally creepy , yet compelling and sad . jeremy renner gives a career-making performance . ",
 1)
```

From the subjective examples above we can see both the sentences are either opinions or suggestions that a person feels. Hence, we can conclude they are subjective examples.

Q3.1.2 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-
Subjective label-

ice age is the first computer-generated feature cartoon to feel like other movies, and that makes for some glacial pacing early on .

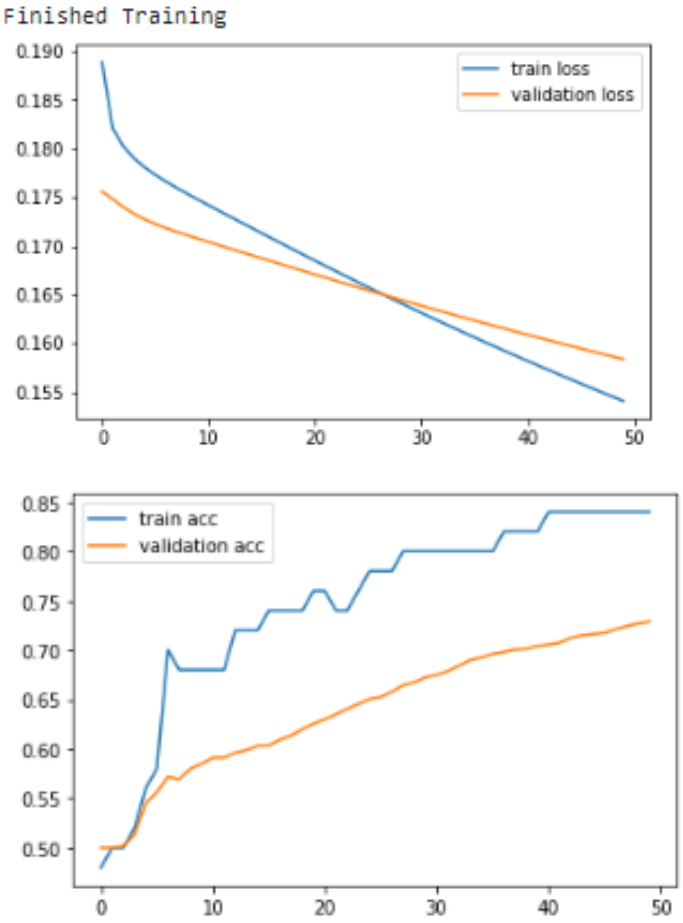
Even though the above sentence is labeled subjective I feel it is objective as Ice age is the first computer generated feature cartoon that has a natural feel.

Objective label-

their love for each other is like a beacon in the storm.

The sentence above is labeled as objective but this can be an opinion as there is no way to measure their love and love is very subjective.

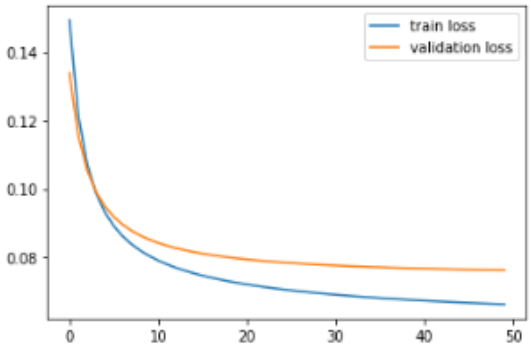
Q4.4 Provide the training loss and accuracy plot for the overfit data in your Report.

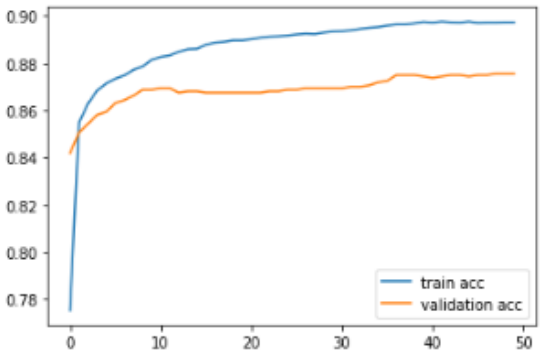


These plots are creatred with the overfit dataset over 50 epochs with a batch size of 4. The X-axis is epochs and the y axis is error and accuracy respectivly. We can see from both these plots that the baseline model is tending to overfit as the train error is less than validation error in the end and the accuracy of the training is much higher than the validation. A accuracy of 85% was achieved on the overfit dataset.

Q4.5 Give 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. Answer this questions: In the baseline model, what information contained in the original sentence is being ignored?

A-
Loss and accuracy curves






```
test error= 0.06724662682181225
test_accuracy 0.9005
```

Final test accuracy- 90%

In the baseline model the order of the words in the sentence is being ignored as the embeddings are all averaged together.

Q4.6 Use that function to determine the 20 closest words to that vector. You should see some words that make it clear what the classifier is doing. Do some of the words that you generated make sense? Explain.

A-

```
[13] print_closest_cosine_words(model.out.weight.squeeze(),20)
```

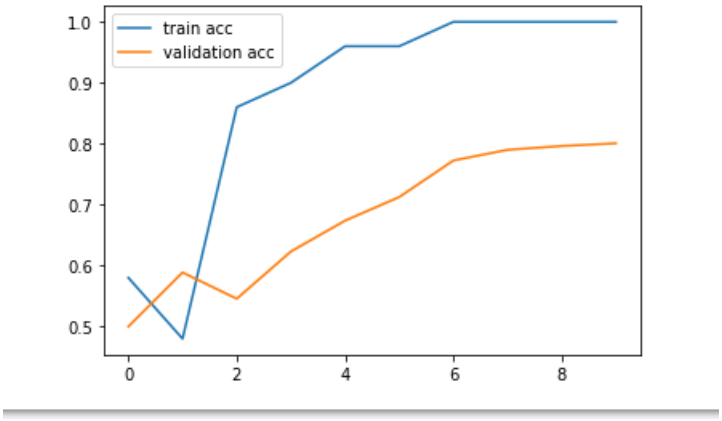
```
flattering      0.53
pleasing        0.52
watchable       0.52
campy   0.51
underwhelming   0.51
verbiage        0.51
emptier         0.49
simplistic      0.49
appetizing      0.49
funnily         0.49
laughable       0.48
tepid   0.48
visuals         0.48
goofier         0.48
mesmerizing     0.48
nuanced         0.48
laudatory       0.47
glib    0.47
befuddling      0.47
forgettable     0.47
```

The weights of the model can be considered a embedding of its own and the purpose of the weights is to predict if the sentence is objective or subjective. If the sentence is subjective the model outputs a 1, hence we can say the weights have a positive response. This response can be seen in the words it is closest to. A subjective sentence tends to be **flattering/ pleasing** as

there are many compliments in the movie reviews. The reviews can also be **nuanced** and provided vivid **visuals** in the sentences. We conclude that these words try to describe subjective sentences.

Q5.2 use the overfit dataset, and the parameters k1 = 2, n1 = 50, k2 = 4, n2 = 50 to make sure that you can overfit the model, as discussed in Section 4.4. Report the accuracy that you were able to achieve with the overfit dataset.

A-
Accuracy plots-

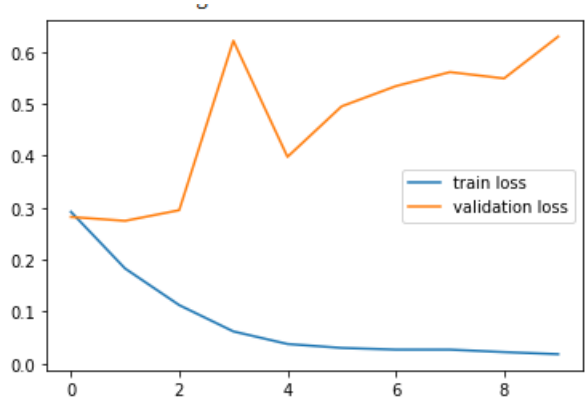


The plot above is the accuracy of the overfit dataset. The Y axis is accuracy, and the X-axis is epochs. It can be seen in 6 epochs the model is able to overfit the dataset with an accuracy of 1 on the overfit dataset.

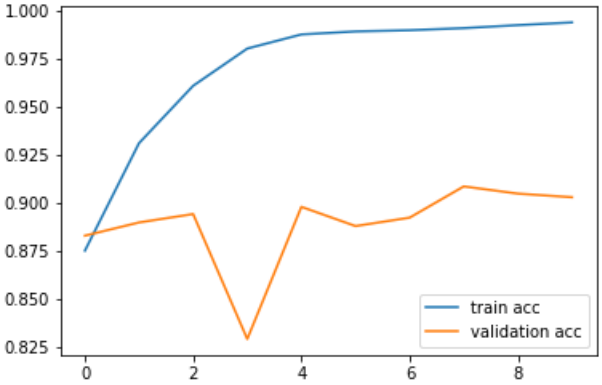
Q5.3.1 Here you should explore the normal hyper parameters for neural networks along with the specific ones in this CNN - k1, n1, k2 and n2. As a suggestion, start with k1 = 2, n1 = 10, k2 = 4, n2 = 10 and select the other hyperparameters. After that, explore different values of k1, n1, k2, n2 to achieve the best accuracy that you can. 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.

A-
Using default parameters-

Losses-



Accuracy-



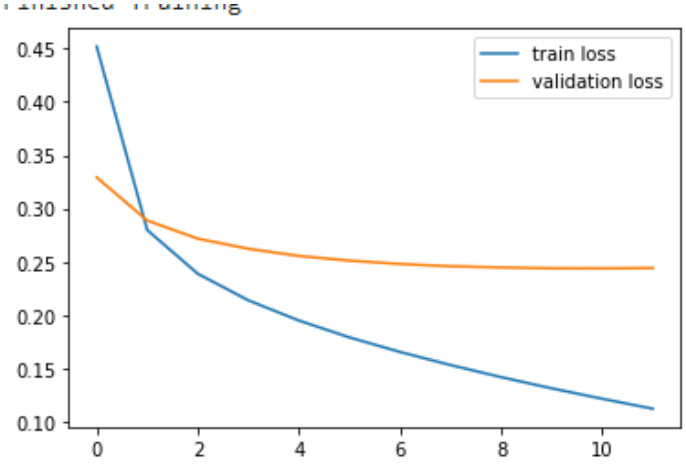
Test set-

```
test error= 0.5337222434529215
test_accuracy 0.9085
```

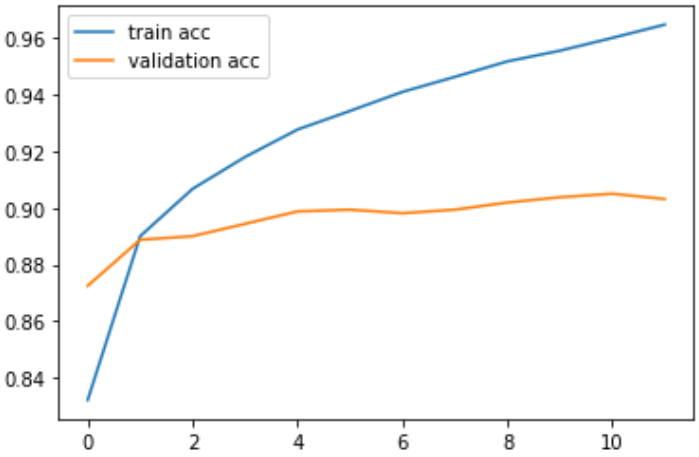
From the plots above we can see the model is overfitting, hence for the tunning process we reduced the expressivity of the model by reducing the number of kernel. We can also reduce the learning rate to avoid the large deviations in the plot above. Dropout can also be implemented to increase the generalization of the model.

After tuning-
The best output after testing was n1=n2=25, k1=2,k2=3, learning rate= 0.0001 and no dropout.

Losses-



Accuracy-



Test set-

```
test error= 0.21894660845937391
test_accuracy 0.9115
```

From the plots above we can see the model is much more stable it has a lower validation loss and slightly higher validation accuracy. It also performs better on the test set.

Q5.3.2 Re-run your best model, but allow the embeddings to be fine-tuned during the training, by setting the freeze parameter to False on the nn.Embedding.from_pretrained class. Report the accuracy of the result, and comment on the result. Save this model in a .pt file

A-

The code for this was completed but running this on CPU was taking extremely long and the outputs were not produced due to this extremely long run time. But you can refer to the code for the procedure.

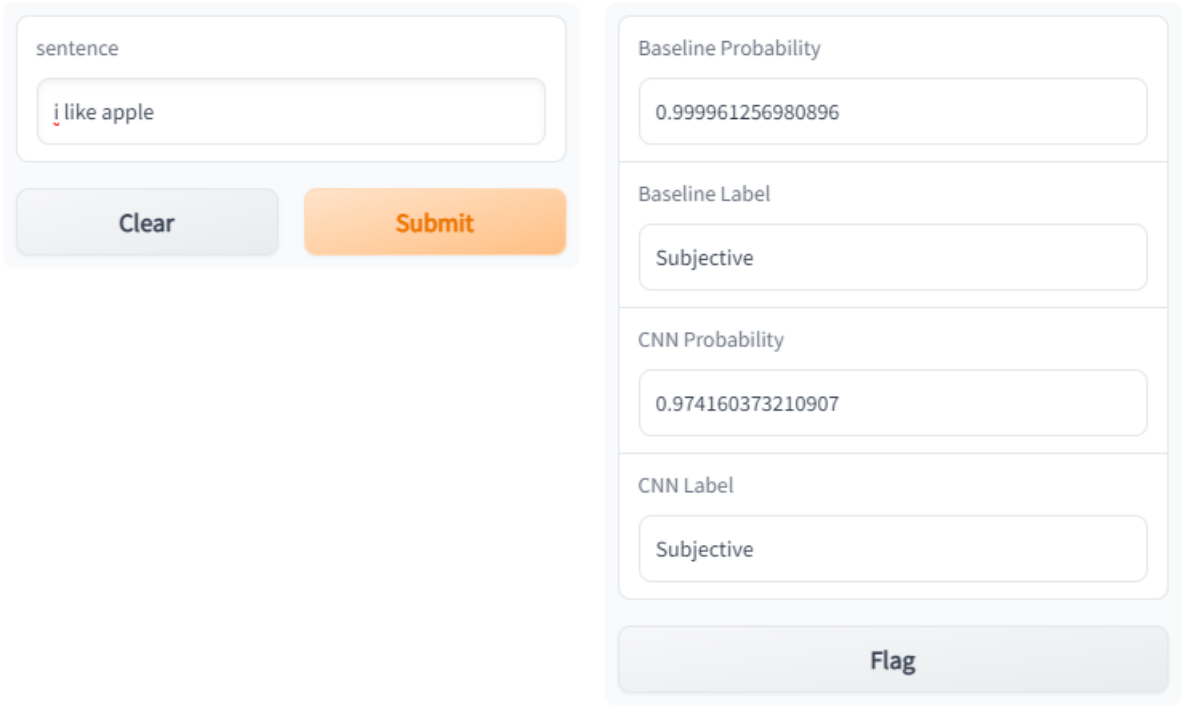
Q5.4 You can explore that meaning using the function print_closest_cosine_words from Assignment 1. Use that function to determine the five closest words to each of the words in the the kernels trained in your best classifier. Do those words make sense? Do the set of words in each given kernel give a broader insight into what the model is looking for? Explain.

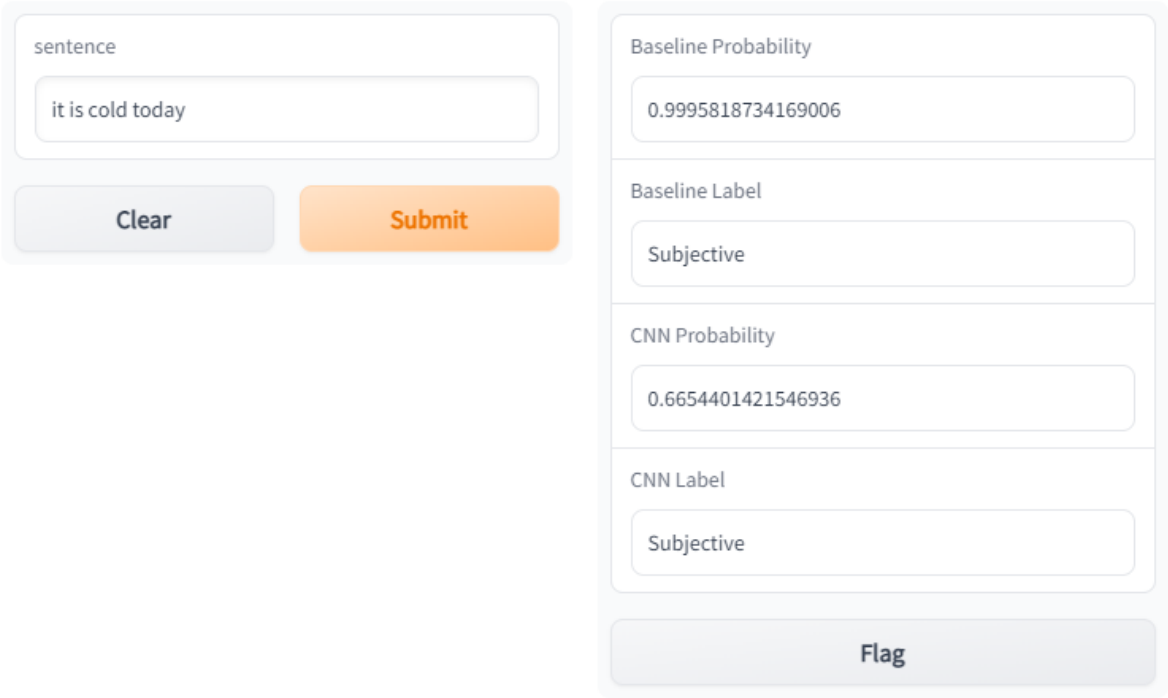
A-

```
kernal- 0
their    0.48
schoolmates      0.45
mid-teens        0.44
classmates       0.44
returned         0.44
kernal- 1
kilkenny         0.42
pupate   0.38
offaly   0.38
their    0.38
capitalise       0.38
kernal- 2
kusuo    0.49
neena    0.49
sunita   0.48
luza     0.47
martorana        0.46
kernal- 3
cashed   0.44
returned         0.43
recontest        0.43
sabbatical       0.43
livings          0.42
kernal- 4
letterboxed      0.47
restrictively    0.47
outselling       0.47
flatteringly     0.44
digitised        0.44
kernal- 5
art-1stld        0.39
activists        0.39
pyrams   0.38
intelligencer    0.36
boersen          0.36
```

From the results above we can see that the CNN model is more complex and the meanings of the kernels don't directly relate with whether a sentence is objective or subjective the way it was in the baseline model. But what we can understand is that each kernel is looking for a particular characteristic eg. Kernel 3 is looking for something that was cashed in or returned and all the words have to do with money. The model then uses each of these characteristics to classify the sentence.

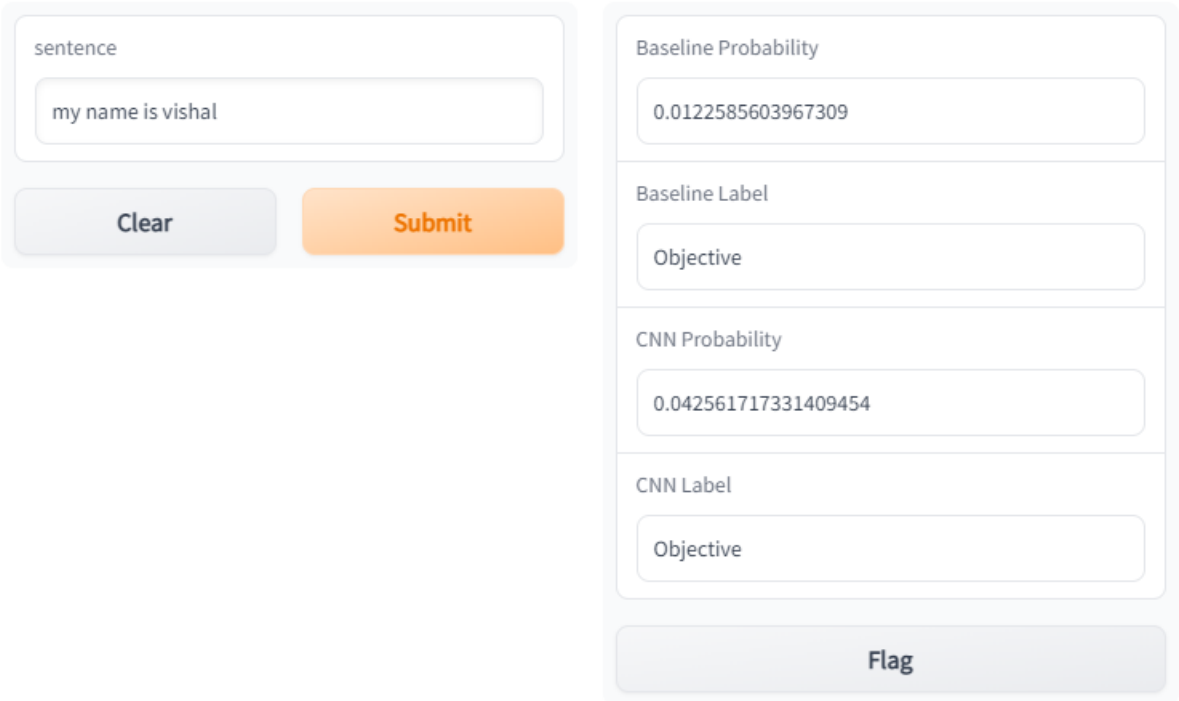
Q6.1 Run your two best stored models on 4 sentences that you come up with yourself, where two of the sentences are definitely objective/subjective, and the other two are borderline subjective/objective, according to your opinion. Include the input and output in your write up. Comment on how the two models performed and whether they are behaving as you expected. Do they agree with each other? Which model seems to be performing the best?
A-
Subjective examples

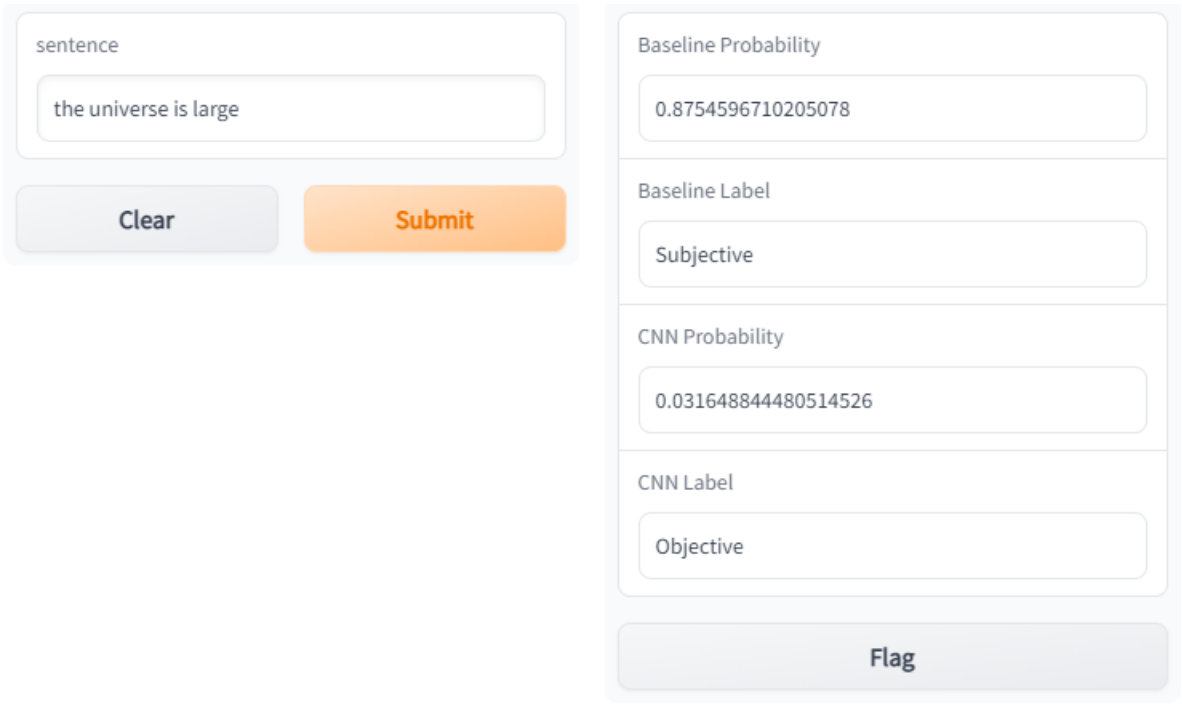




We can see “I like apples” is clearly subjective and both models do well. But for “ it is cold today” it can be interpreted as an objective sentence too as cold can be a fact or it can be subjective to the person and the CNN model picks this up.

Objective examples-





“my name is vishal” is objective and both models do well in identifying this. “The universe is large” is also kind of an edge case as large is vague but at the end it is still a fact. The CNN does pick this up while the baseline fails.
Overall the CNN performs very well.