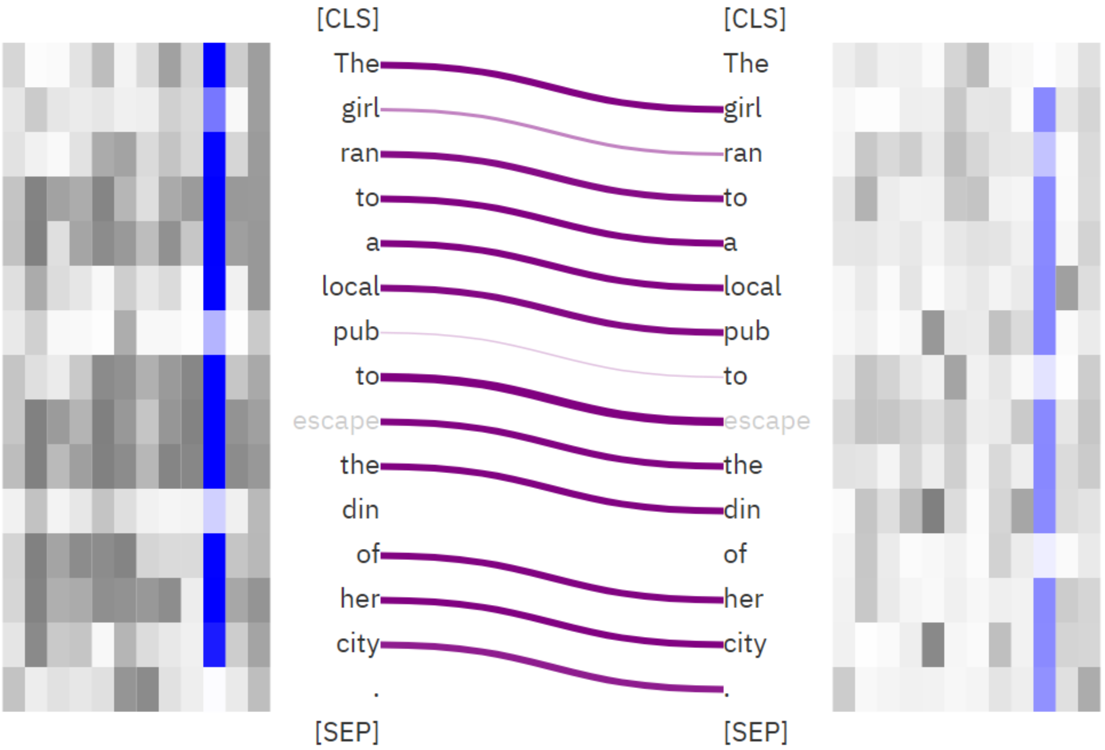
1. Describe your understanding and findings about the **attention mechanism** by exBERT. (20%)

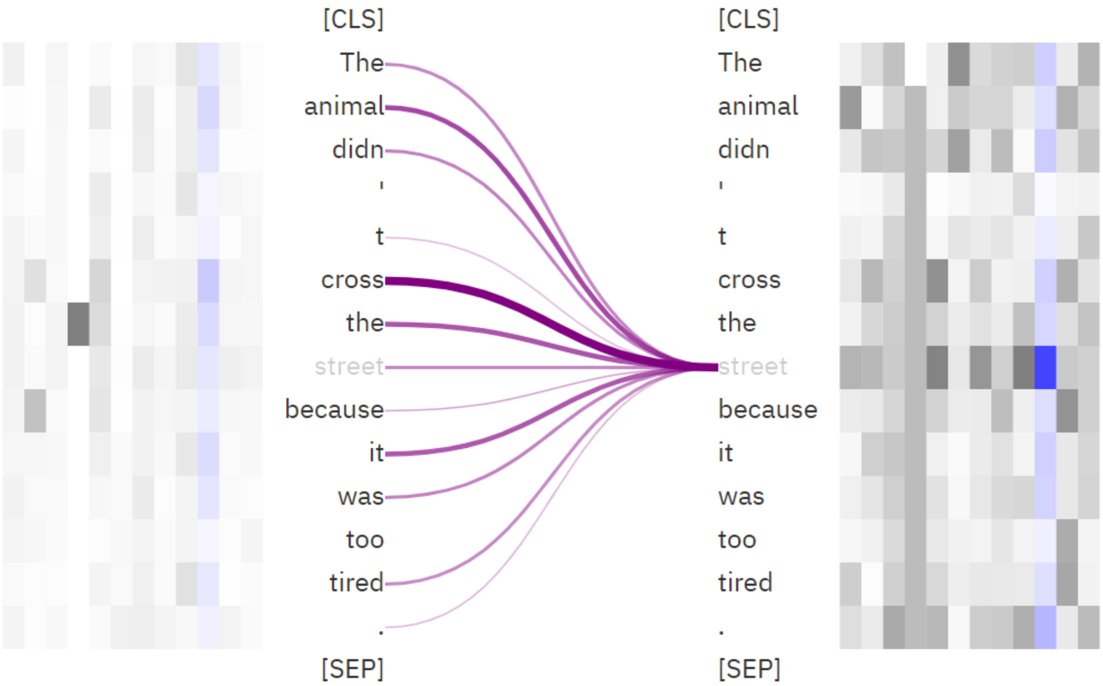
Ans: In BERT, the attention mechanism is used to compute attention weights between each pair of tokens in the input sequence. These attention weights determine the importance or relevance of each token to other tokens in the sequence. The attention weights are computed through a process known as self-attention, where the model attends to different positions in the input sequence to gather information. ExBERT investigates the attention patterns learned by BERT and explores how they contribute to model predictions. It seeks to uncover the reasoning and decision-making processes of BERT by analyzing the attention weights assigned to different tokens.

By adjusting parameters on the website Exbert - a Hugging Face Space by exbert-project, we can observe the attention visualization. I mainly adjust the number of layers and selected heads to see the difference in visualization pattern. The layer index in exBERT ranges from 0 to the total number of layers in the BERT model. On the other hand, in a multi-head attention mechanism, the input is processed through multiple parallel attention heads, each responsible for learning different patterns and capturing different types of dependencies or relationships, enabling the model to focus on different aspects of the context simultaneously.

For example (1), the input sequence is “The girl ran to a local pub to escape the din of her city.” Choose layer = 7 and selected head = 10, we can see that the masked word to be predicted is in high relevance with its former latter (or mark). Another example (2) with input sequence is “The animal didn't cross the street because it was too tired.” Choose layer = 3 and selected head = 10 and we focus on the masked word “street”. We can see that “cross” contributes the most (since its weight is the highest) in predicting the missing word.



▲ (1)

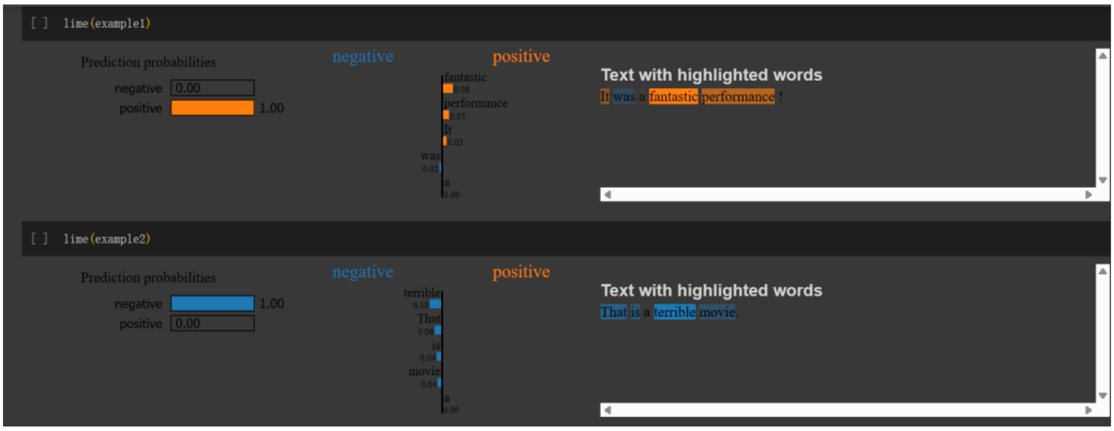


▲ (2)

2. Compare at least **2 sentiment classification models** (e.g., TA_model_1.pt, your model in HW2). (30%)

Ans:

```
example1 = 'It was a fantastic performance !'
example2 = 'That is a terrible movie.'
Example3 = 'I am so happy and surprised that there is so much
interest in this movie!'
Example4 = 'Oh, my goodness. I would have never thought it was
possible for me to see a thriller worse than Domestic Disturbance
this soon, but here it is.'
```



```
[ ] lime(example1)
```

Text with highlighted words

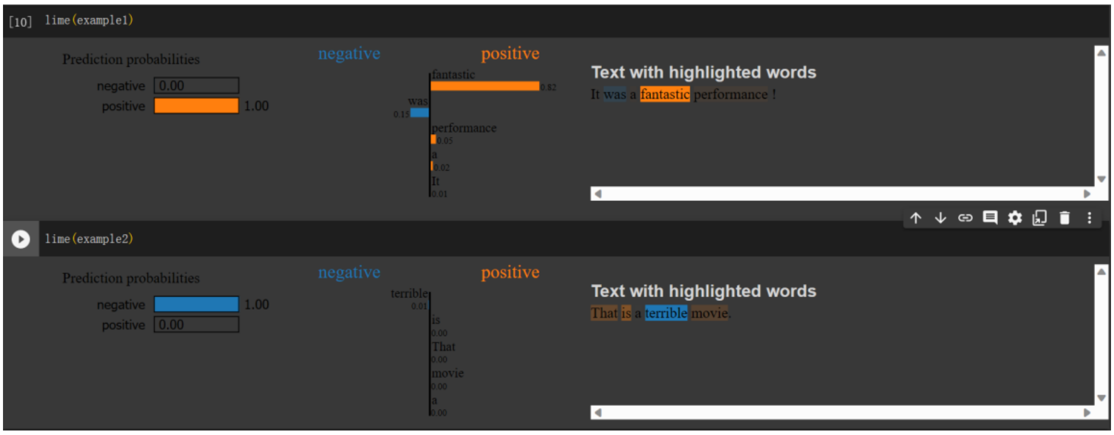
It was a fantastic performance !

```
[ ] lime(example2)
```

Text with highlighted words

That is a terrible movie.

▲ LIME for TA_model_1.pt with example1 and example2



```
[10] lime(example1)
```

Text with highlighted words

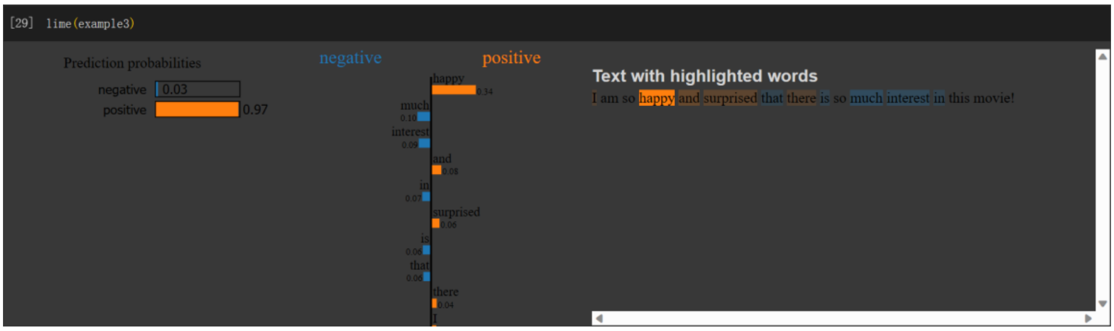
It was a fantastic performance !

```
lime(example2)
```

Text with highlighted words

That is a terrible movie.

▲ LIME for TA_model_2.pt with example1 and example2

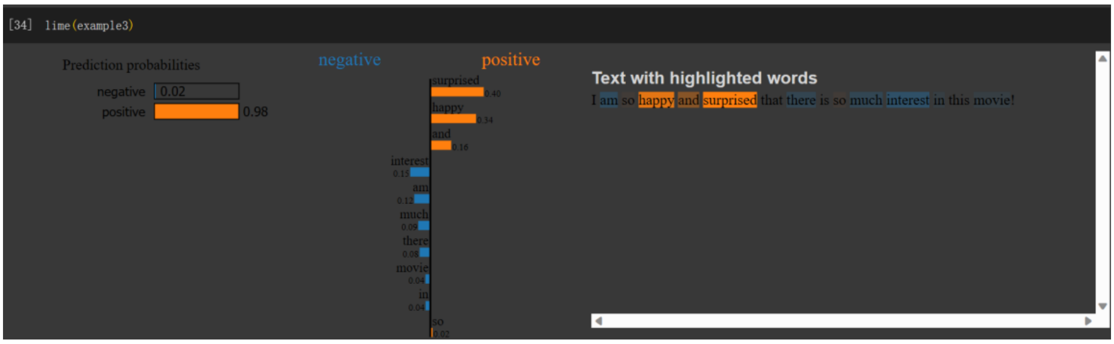


```
[29] lime(example3)
```

Text with highlighted words

I am so happy and surprised that there is so much interest in this movie!

▲ LIME for TA_model_1.pt with example3

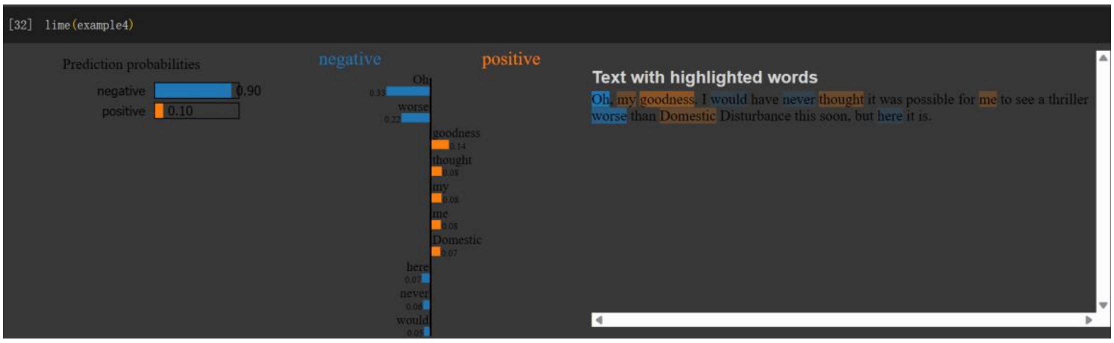


```
[34] lime(example3)
```

Text with highlighted words

I am so happy and surprised that there is so much interest in this movie!

▲ LIME for TA_model_2.pt with example3

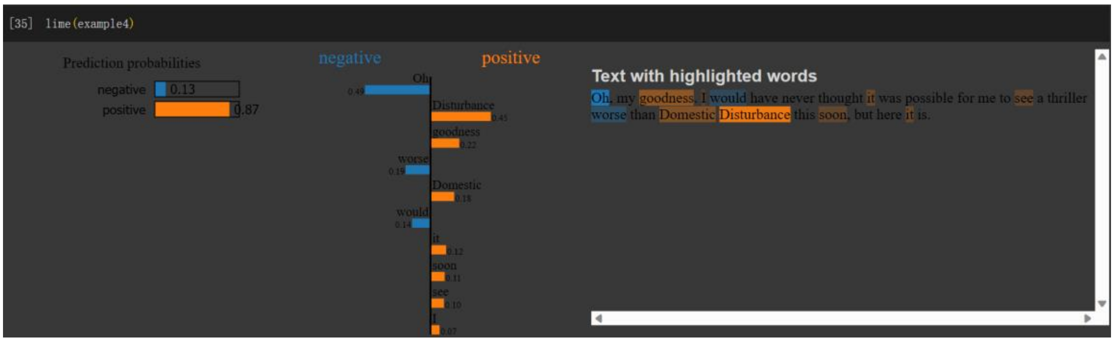


```
[32] lime(example4)
```

Text with highlighted words

Oh, my goodness. I would have never thought it was possible for me to see a thriller worse than Domestic Disturbance this soon, but here it is.

▲ LIME for TA_model_1.pt with example4



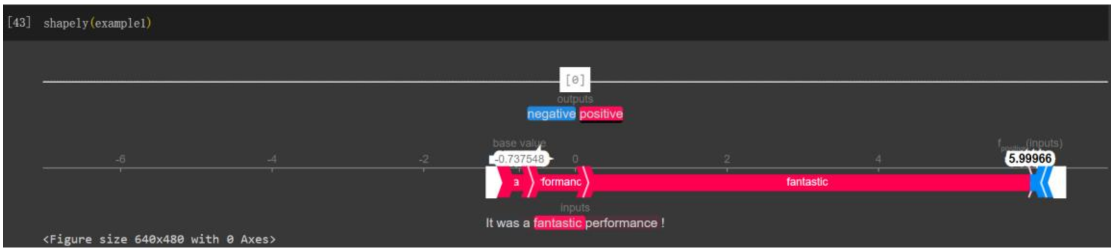
```
[35] lime(example4)
```

Text with highlighted words

Oh, my goodness. I would have never thought it was possible for me to see a thriller worse than Domestic Disturbance this soon, but here it is.

▲ LIME for TA_model_2.pt with example4

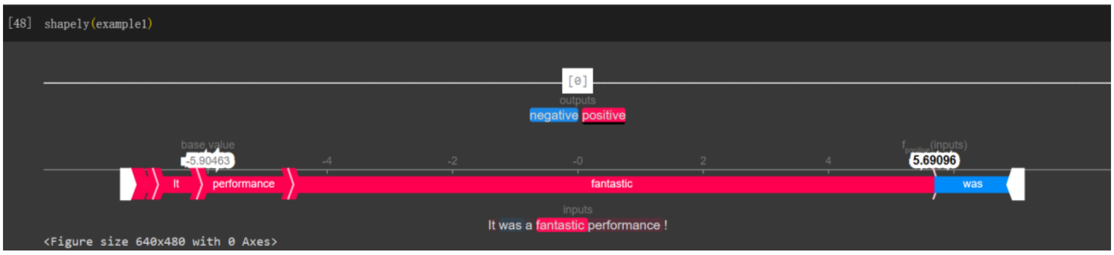
Applying LIME, from the above examples, I'll make a conclusion that TA_model_2.pt will boldly give higher score for words it think that are relevant to a positive feedback, such as 'surprised' – 0.06 in TA_model_1.pt but 0.40 in TA_model_2.pt, and 'fantastic' – 0.08 in TA_model_1.pt but 0.82 in TA_model_2.pt. It may lead to type II error (False Positive). We can see FP happens for TA_model_2.pt in example4 since it wrongly gives a high score for 'Disturbance' in relation to positive.



```
[43] shapely(example1)
```

```
<Figure size 640x480 with 0 Axes>
```

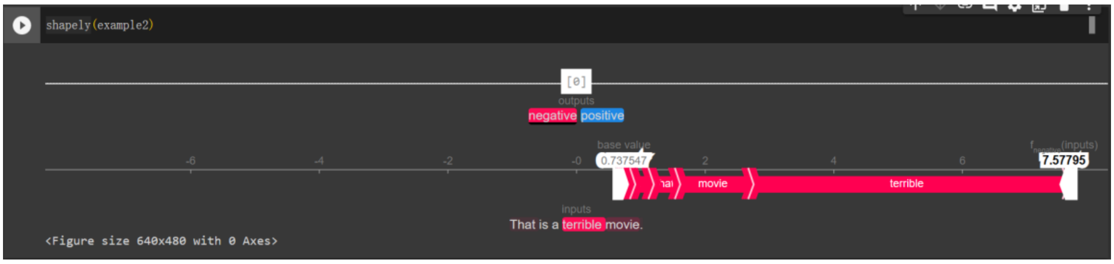
▲ SHAP for TA_model_1.pt with example1



```
[48] shapely(example1)
```

```
<Figure size 640x480 with 0 Axes>
```

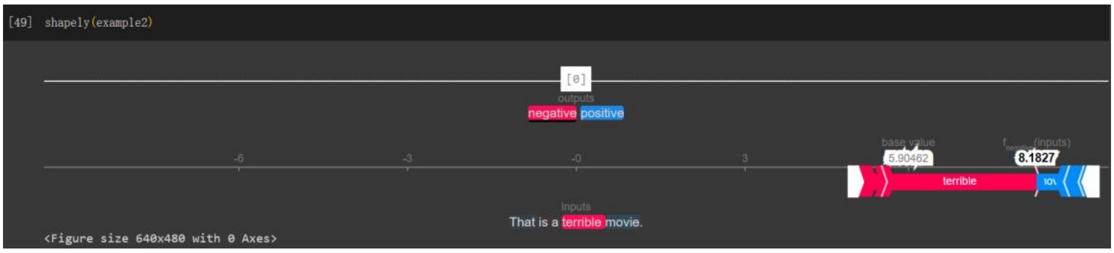
▲ SHAP for TA_model_2.pt with example1



```
shapely(example2)
```

```
<Figure size 640x480 with 0 Axes>
```

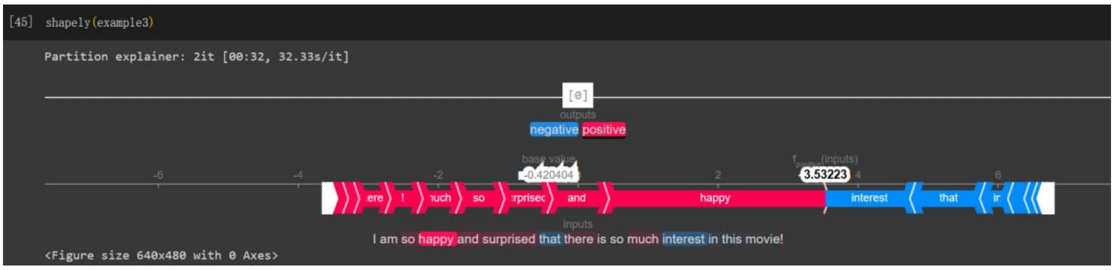
▲ SHAP for TA_model_1.pt with example2



```
[49] shapely(example2)
```

```
<Figure size 640x480 with 0 Axes>
```

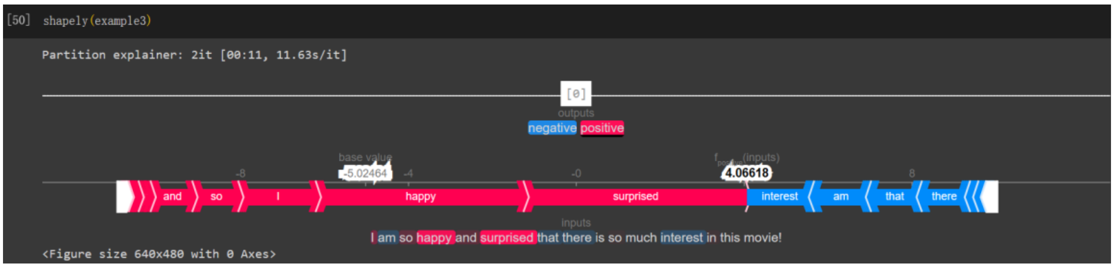
▲ SHAP for TA_model_2.pt with example2



```
[45] shapely(example3)

Partition explainer: 2it [00:32, 32.33s/it]
```

```
I am so happy and surprised that there is so much interest in this movie!
<Figure size 640x480 with 0 Axes>
```

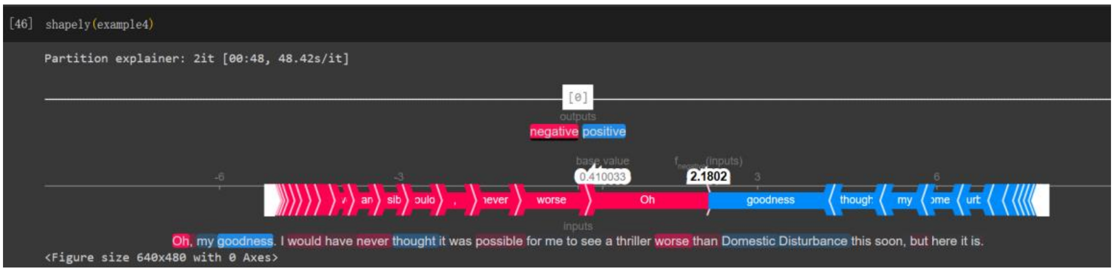
▲ SHAP for TA_model_1.pt with example3



```
[50] shapely(example3)

Partition explainer: 2it [00:11, 11.63s/it]
```

```
I am so happy and surprised that there is so much interest in this movie!
<Figure size 640x480 with 0 Axes>
```

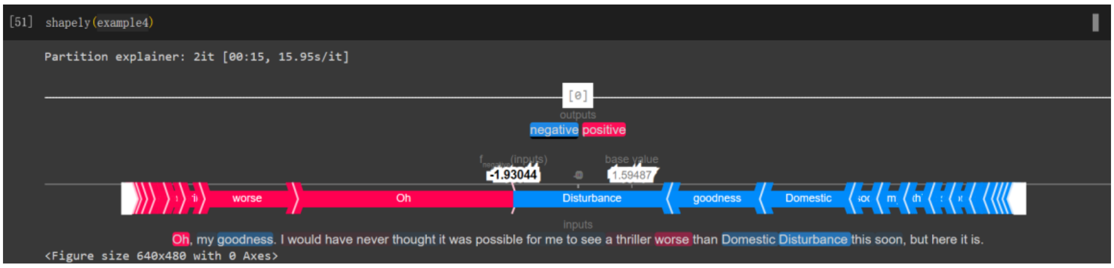
▲ SHAP for TA_model_2.pt with example3



```
[46] shapely(example4)

Partition explainer: 2it [00:48, 48.42s/it]
```

```
Oh, my goodness. I would have never thought it was possible for me to see a thriller worse than Domestic Disturbance this soon, but here it is.
<Figure size 640x480 with 0 Axes>
```

▲ SHAP for TA_model_1.pt with example4



```
[51] shapely(example4)

Partition explainer: 2it [00:15, 15.95s/it]
```

```
Oh, my goodness. I would have never thought it was possible for me to see a thriller worse than Domestic Disturbance this soon, but here it is.
<Figure size 640x480 with 0 Axes>
```

▲ SHAP for TA_model_2.pt with example4

Except the fact that we discuss in applying LIME, by applying SHAP, we can conclude that TA_model_2.pt would have a lower base value for positive output and a higher base value for negative output, which coincides with our guess that TA_model_2.pt tend to False Positive.

3. Compare the explanation of **LIME and SHAP**. (30%)

Ans: LIME creates local explanations by approximating the behavior of the black-box model around the prediction of interest. To be more specific, we sample and perturb the input features, training an interpretable model (e.g., a linear model) on these perturbed instances to approximate the behavior of the model locally. SHAP provides global explanations by leveraging game theory concepts to assign feature importance values. It calculates the contribution of each feature to the prediction outcome by considering all possible combinations of features and comparing the model's predictions with and without each feature.

Next we compare the output of the two different methods. LIME provides explanations in the form of feature importance weights for the input variables, indicating their influence on a particular prediction. On the contrary, SHAP provides feature attributions, which represent the contribution of each feature to the prediction outcome. It produces a set of Shapley values, where each feature's importance is quantified. We can see their difference with the results in problem 2. SHAP not only provides the contribution (importance) of each word in a sentence, but also has a base value to ensure that the predictions won’t be only based on the weighted sum of positive and negative feedback, providing a more comprehensive assessment.

In summary, LIME focuses on generating local explanations for individual predictions using perturbation-based methods, while SHAP provides both local and global explanations by assigning feature importance values based on game theory concepts.

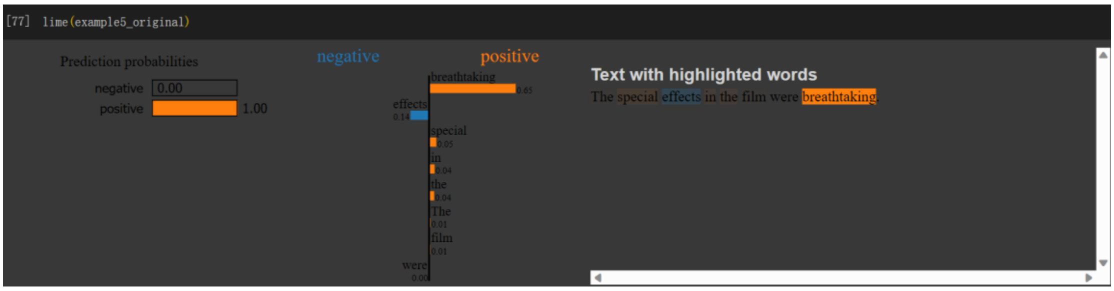
4. Describe how you implement other explanation techniques. And discuss with the explanation result. (Bonus)

Ans: Skip this part.

5. Try 3 different input sentences for **attacks**. Also, describe your findings and how to prevent the attack if you retrain the model in the future. (20%)

Ans:

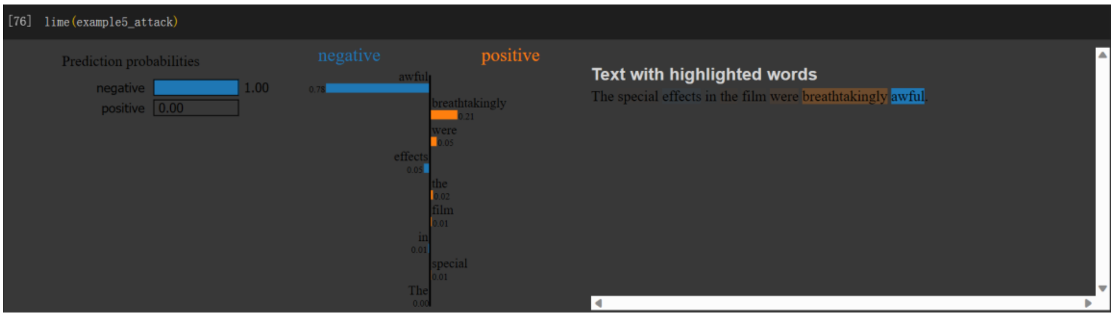
```
example5_original = "The special effects in the film were breathtaking."
example5_attack = "The special effects in the film were breathtakingly awful."

example6_original = "I love this movie."
example6_attack = "I loooooove this movie."

example7_original = "I read all of the other comments which made this movie out to be an excellent movie. I saw nothing of the excellence that was stated."
example7_attack = "I read all of the other comments which made this movie out to be an EXCELLECT movie. I saw NOTHING of the excellence that was stated."
```



```
[77] lime(example5_original)
```

Text with highlighted words

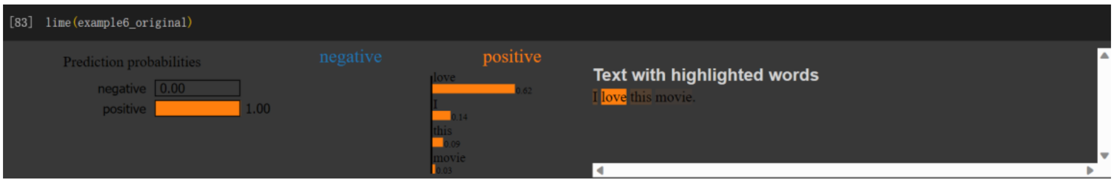
The special effects in the film were breathtaking.



```
[76] lime(example5_attack)
```

Text with highlighted words

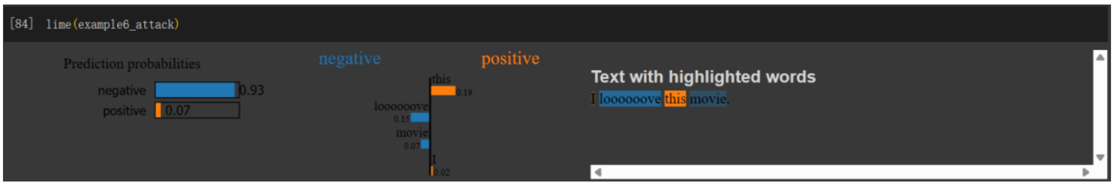
The special effects in the film were breathtakingly awful.



```
[83] lime(example6_original)
```

Text with highlighted words

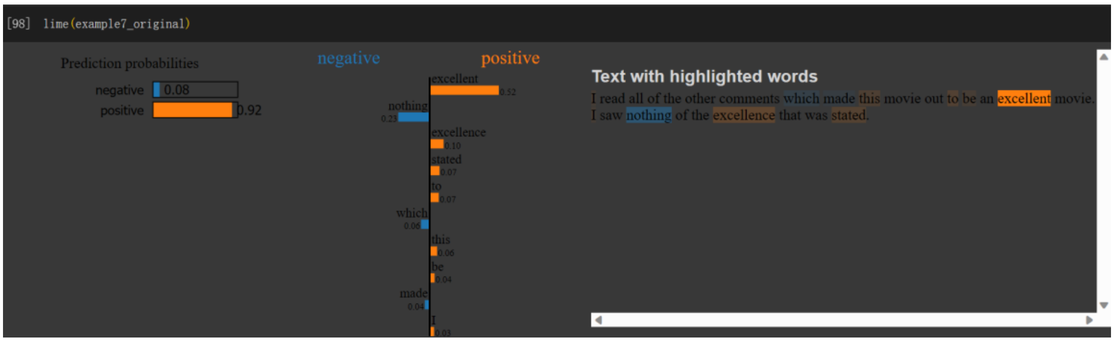
I love this movie.



```
[84] lime(example6_attack)
```

Text with highlighted words

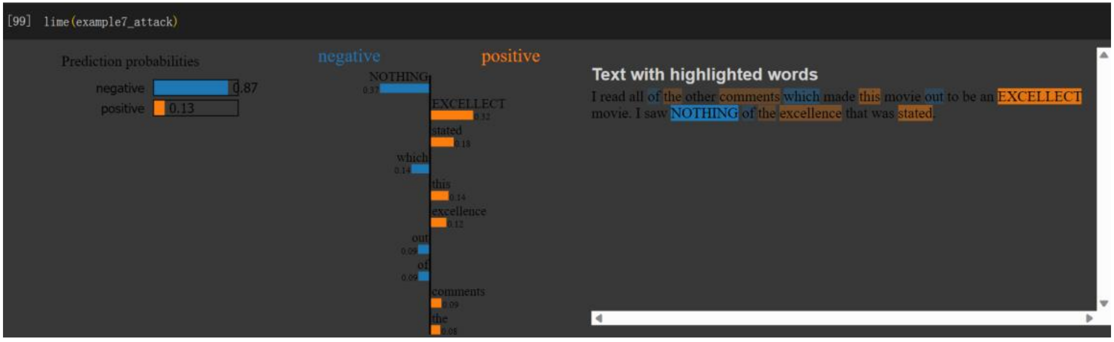
I looooooove this movie.



```
[98] lime(example7_original)
```

Text with highlighted words

I read all of the other comments which made this movie out to be an excellent movie. I saw nothing of the excellence that was stated.



```
[99] lime(example7_attack)
```

Text with highlighted words

I read all of the other comments which made this movie out to be an EXCELLECT movie. I saw NOTHING of the excellence that was stated.

I found that some words which have multiple meanings will lead the model to confusion. For example, "awful" can represent "dreadful", or it can mean that something is extremely bad. In example 5, "breathtakingly awful" should be a fair assessment, but our model detected "awful" and classified it into a negative review. To fix this problem, we can adjust our model to make it more sensitive when assessing a multi-meaning word, giving more weight (attention) to other words in the context. Besides, we can transform the upper-case word into lower-case and remove the unseen words (perhaps misspelling or exaggerating) to get a fair judgment.

6. Describe problems you meet and how you solve them. (Bonus)

Ans: In the beginning, I felt confused about the visualization result and I read many documents or comments, trying to figure out the concepts they represent. For example, in problem 2, I know that the same explanation method would lead to different results due to the structure of different models, but I don’t know how to use the visualization result to better understand the model, or to gain enough information to have a statement. I solve the problem by giving more examples and discussing with my classmates, and thus finish this assignment.