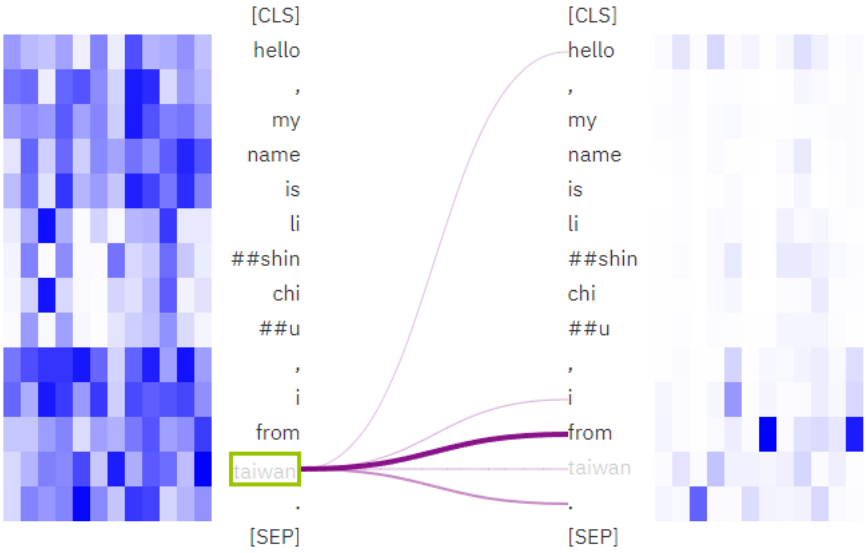
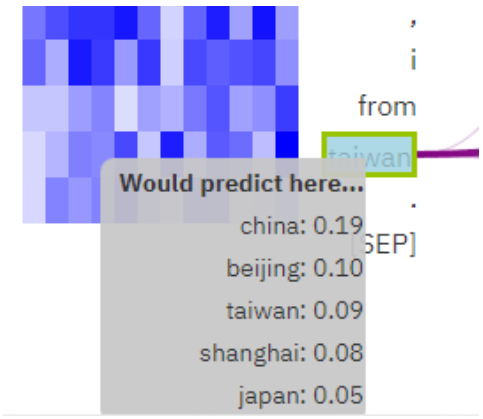
Part 1: Attention Visualization:

Using BERT as Masked Language Model

Using “Hello ,My name is Lishin Chiu,I from Taiwan” and mask the word “Taiwan”.

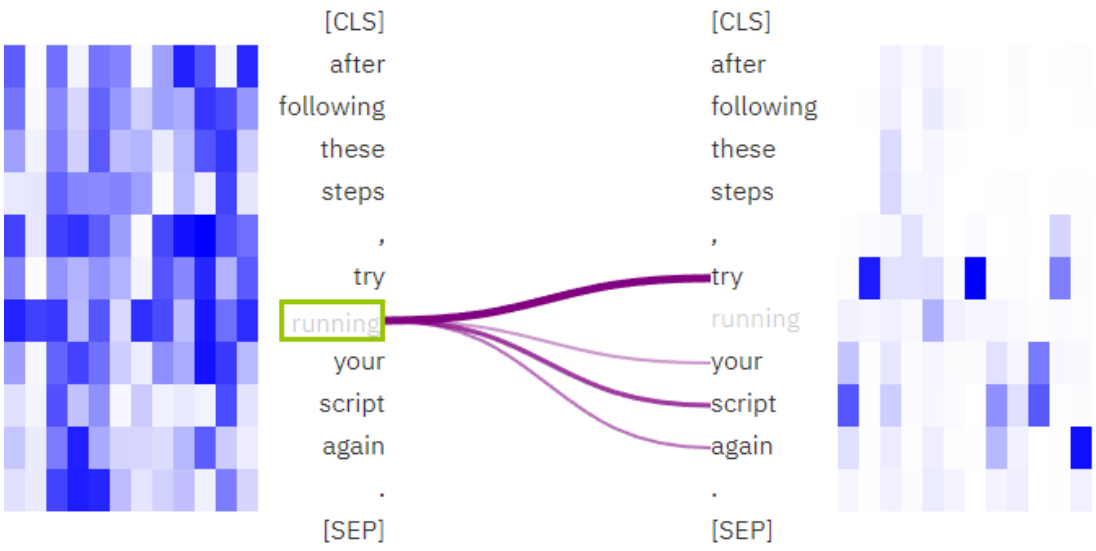


如果我們擋住”Taiwan” 發現它注意的是”Hello:,”I”, ”from”, ”.”

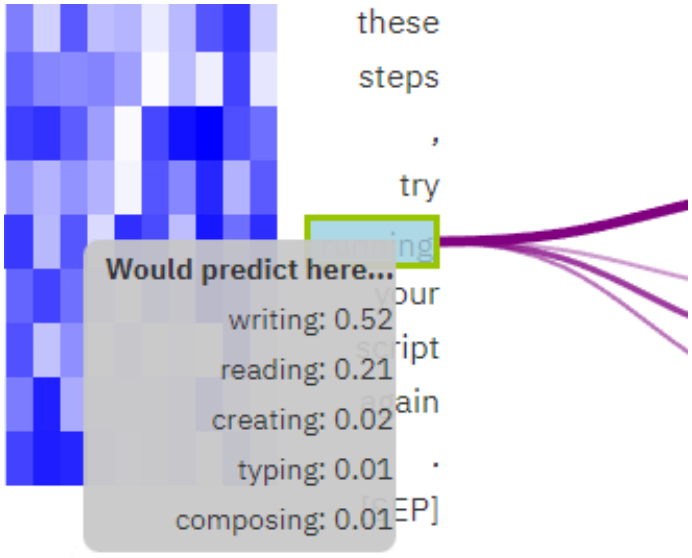


這是 Bert 的猜測，他其實會透過名字來猜地點

Using “After following these steps, try running your script again.”,and mask running



根據圖片，它注意的是”try”,”your”,”script”,”again”



這是 bert 的猜測

我們使用了 distilbert-base-uncased 模型來進行注意力可視化。我們選擇了兩個示例句子：“Hello ,My name is Lishin Chiu,I from Taiwan.” 和 “After following these steps, try running your script again.” ，並使用 exBERT 工具對模型的注意力權重進行了可視化。

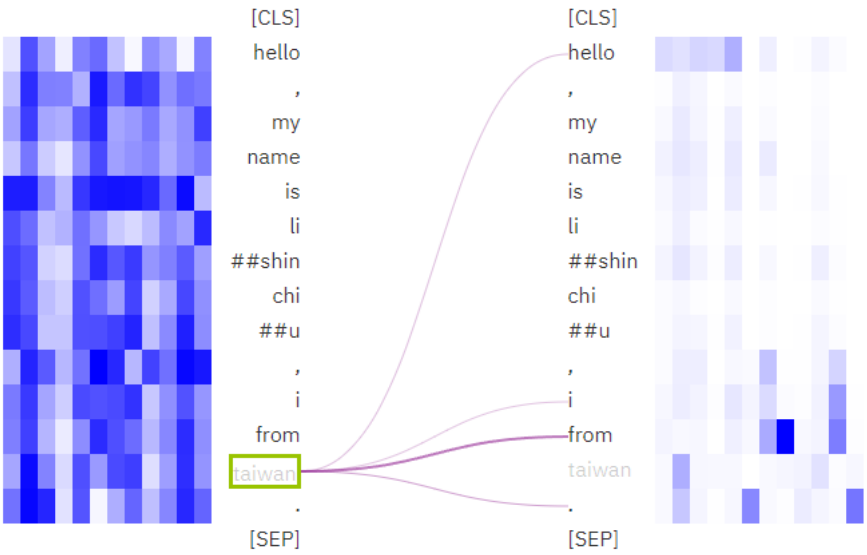
從可視化結果中可以看到，模型對句子中的關鍵詞（例如“from”和“try”）給予了較高的注意力權重。這表明模型在進行分類時，能夠有效地捕捉到這些關鍵詞的重要性。。

總之，上面的例子展示了注意力機制在 BERT 中是如何運作的。與 n-gram ,ELMo 等其他方法相比，這種機制有助於 BERT 表現更好。

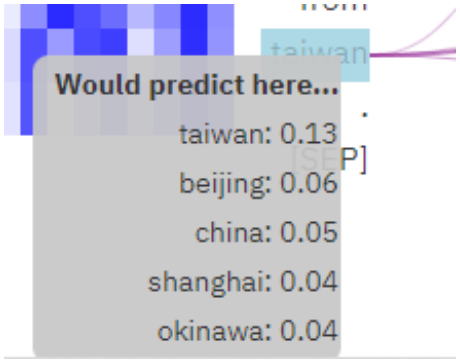
2. Compare at least 2 sentiment classification models(BERT and DistilBERT)

使用和 Part1 相同的句子，不過使用的是 DistilBERT

Using “Hello ,My name is Lishin Chiu,I from Taiwan” and mask the word “Taiwan”.

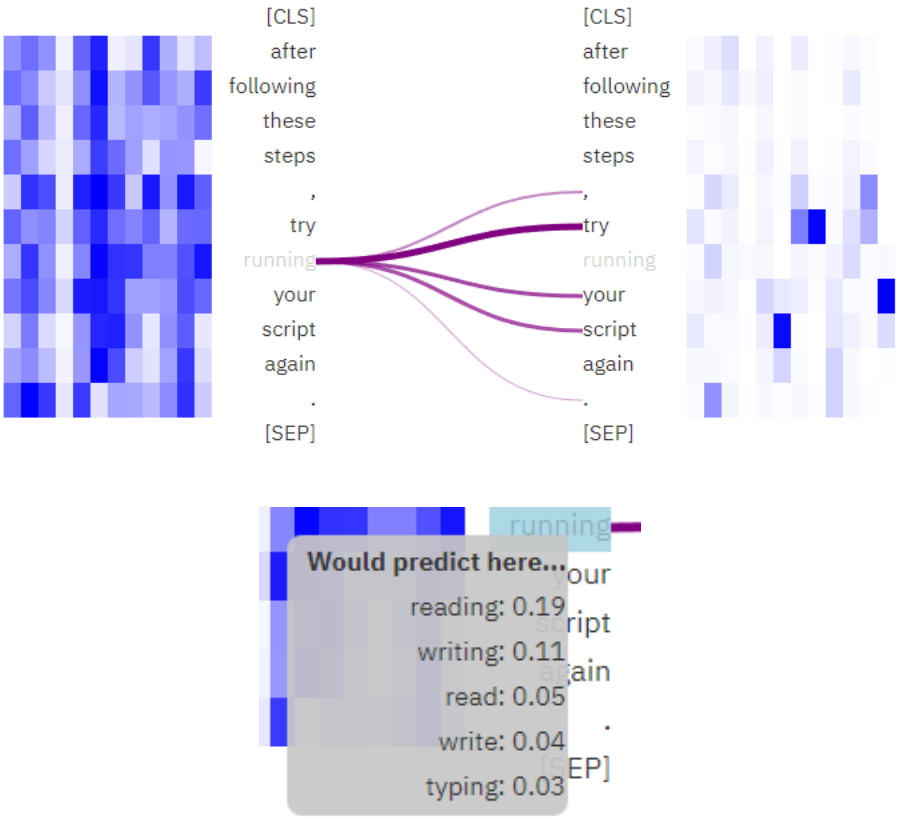


DistilBERT 能夠正確預測，並且它關注的單字與 BERT 相似。這表明 DistilBERT 學習來自原始 BERT 的一些知識。



DistilBERT 的猜測

Using “After following these steps, try running your script again.”,and mask running



DistilBERT 的猜測

發現 DistilBERT 的猜測能力並沒有 BERT 那麼肯定和正確

通過比較不同的預訓練模型，我們發現 distilbert-base-uncased 模型在某些情況下可能會略微忽略一些細節，但整體上能夠保持較高的性能和效率。

3. Compare the explanation of LIME and SHAP

LIME

透過對輸入資料的局部擾動產生新的數據，並觀察這些擾動資料對模型預測的影響，從而訓練一個簡單的、可解釋的模型（如線性模型）來近似原始模型在局部的行為。

它透過取樣產生多個鄰近樣本，並計算每個樣本的模型預測值。然後，它對這些樣本進行加權線性迴歸，以擬合模型在局部的行為。

SHAP

SHAP 基於博弈論中的 Shapley 值，將每個特徵視為一個“玩家”，透過計算每個特徵對預測結果的邊際貢獻，來解釋模型的輸出。

它透過考慮所有可能的特徵組合，並計算每個特徵在不同組合中的邊際貢獻，從而得到每個特徵的 Shapley 值。這些 Shapley 值表示特徵對預測結果的貢獻。

接下來我使用助教給的.ipynb 跑幾種不同測資，來比較 LIME 和 SHAP

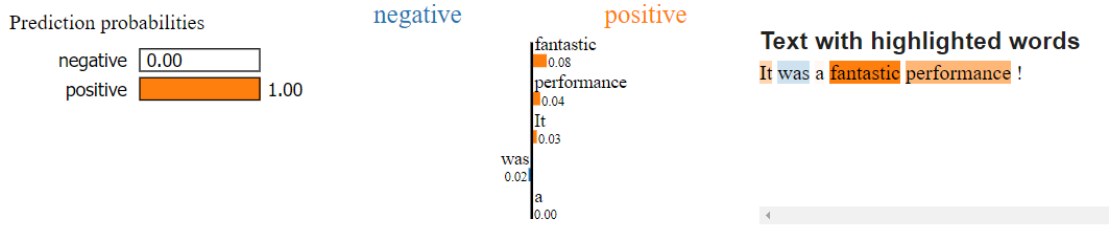
1. It was a fantastic performance!

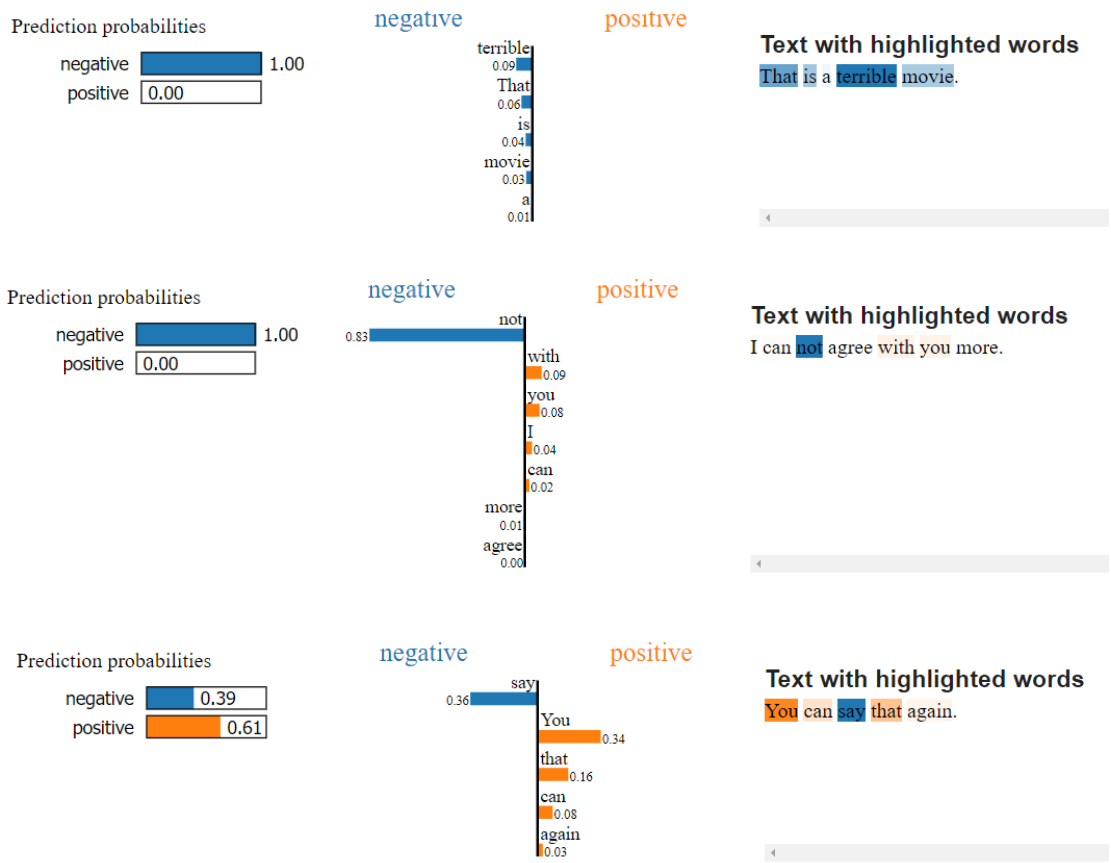
2. That is a terrible movie.

3. I can not agree with you more.

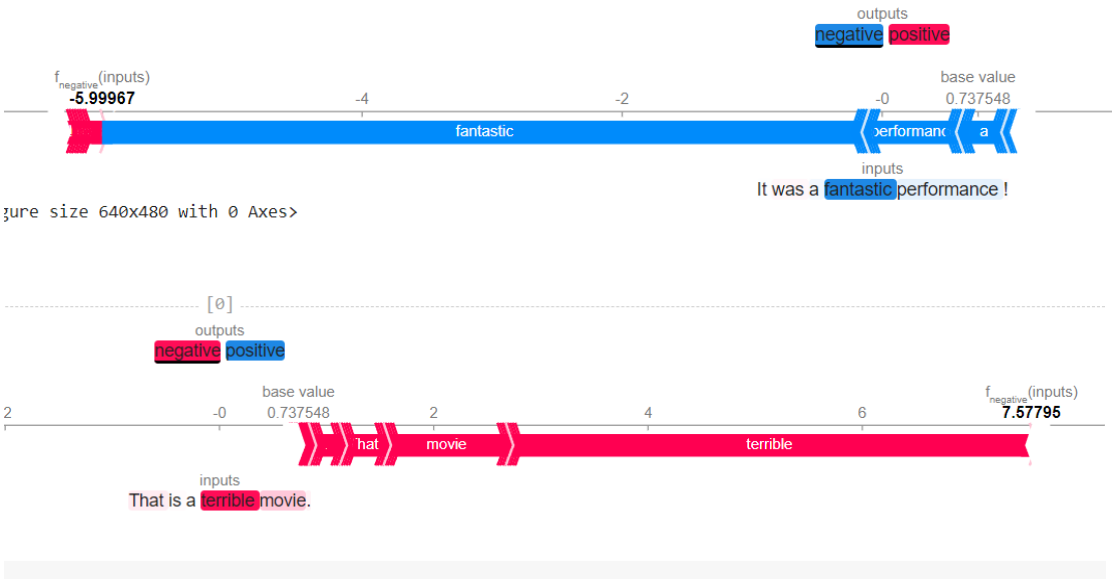
4. You can say that again.

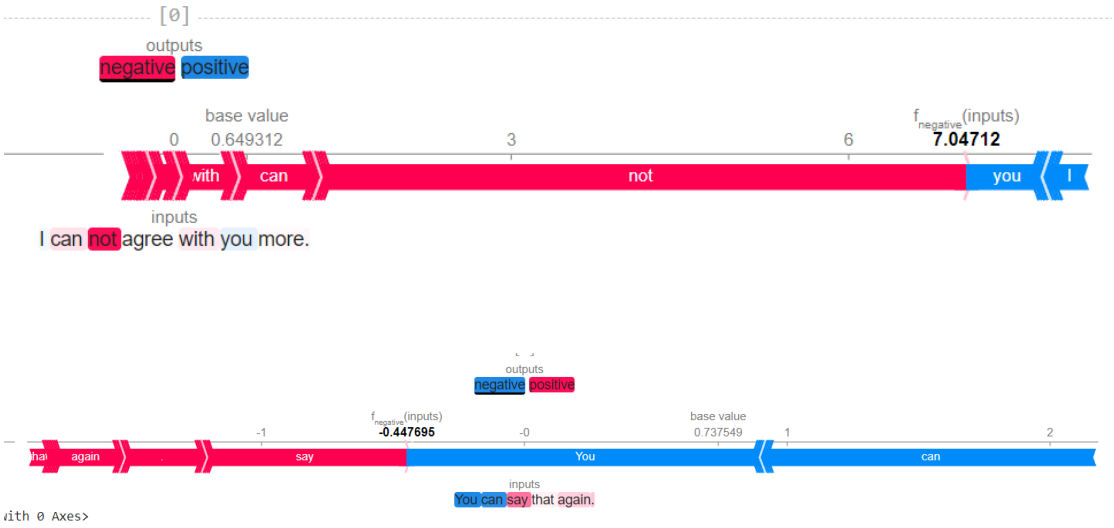
By LIME




Prediction probabilities
negative 1.00
positive 0.00
negative
positive
terrible 0.09
That 0.06
is 0.04
movie 0.03
a 0.01
Text with highlighted words
That is a terrible movie.
Prediction probabilities
negative 1.00
positive 0.00
negative
positive
not 0.83
with 0.09
you 0.08
I 0.04
can 0.02
more 0.01
agree 0.00
Text with highlighted words
I can not agree with you more.
Prediction probabilities
negative 0.39
positive 0.61
negative
positive
say 0.36
You 0.34
that 0.16
can 0.08
again 0.03
Text with highlighted words
You can say that again.

By SMAP


outputs
negative positive
f negative (inputs)
-5.99967
-4
-2
-0
base value
0.737548
fantastic
a
inputs
It was a fantastic performance !
gure size 640x480 with 0 Axes>
[0]
outputs
negative positive
base value
2
-0
0.737548
2
4
6
f negative (inputs)
7.57795
movie
terrible
inputs
That is a terrible movie.

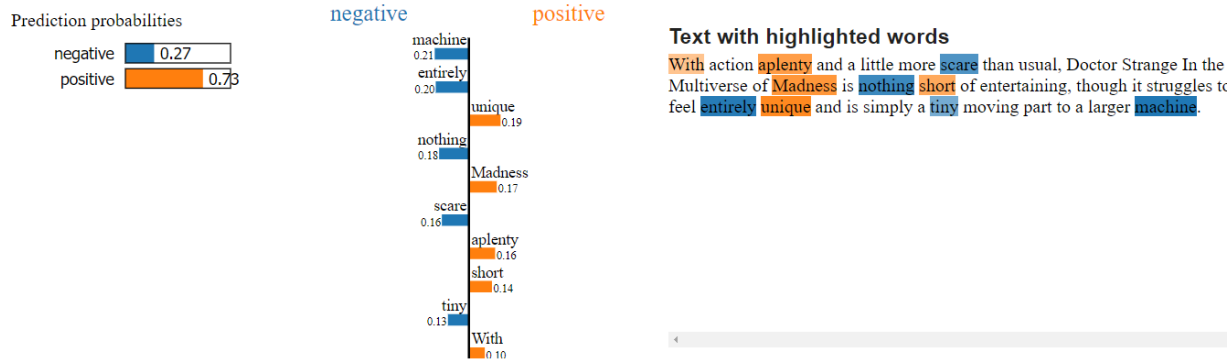


在例子 1、2 中，兩個都能正確的理解句子和句子的意思

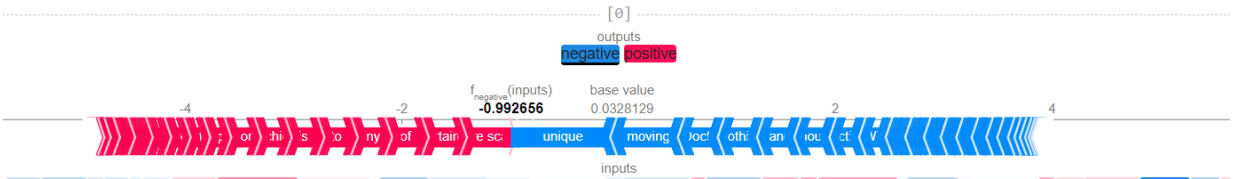
在例子 3、4 中，很明顯是表正面的意思，但它不能完全正確的分析出來，在例子 3 中他們都因為句中的 not 而影響了判斷，在例子 4 中有可能是因為這是經典的句子，所以才分辨得出來

接下來我使用爛番茄中的影評來測試

1. With action aplenty and a little more scare than usual, Doctor Strange In the Multiverse of Madness is nothing short of entertaining, though it struggles to feel entirely unique and is simply a tiny moving part to a larger machine.(給好評)
2. There’s nothing significant in the storytelling of Multiverse of Madness, and Marvel’s new fashion of introducing cameos to bring more audiences to watch these movies has become quite unremarkable.(給差評)

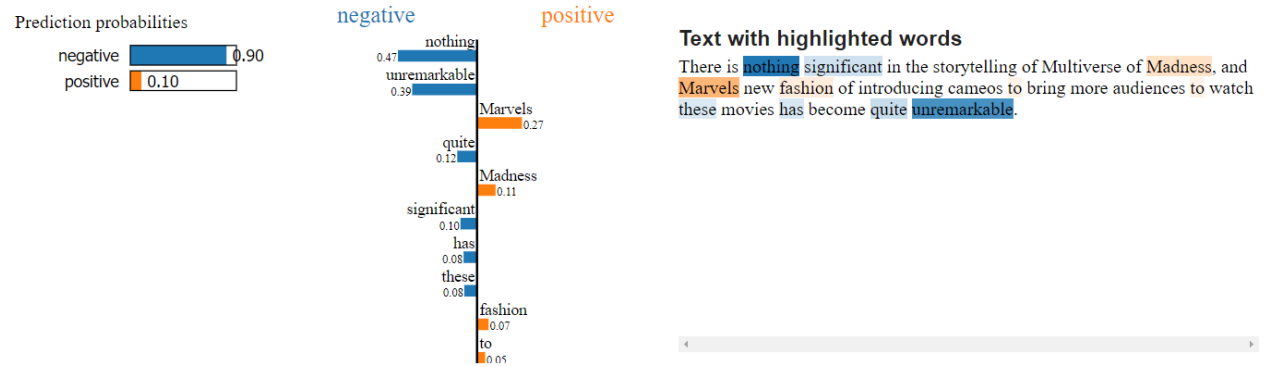


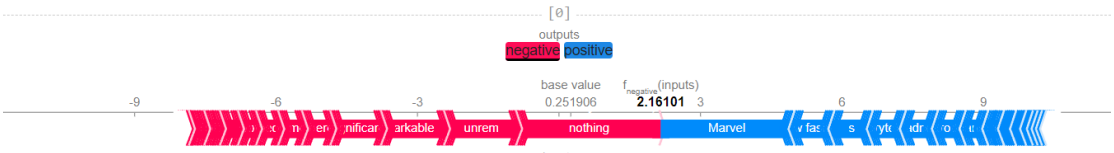

```
  0%|          | 0/498 [00:00<?, ?it/s]
PartitionExplainer explainer: 2it [00:43, 43.02s/it]
```



With action aplenty and a little more scare than usual, Doctor Strange In the Multiverse of Madness is nothing short of entertaining, though it struggles to feel entirely unique and is simply a tiny moving part to a larger machine.

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





There is nothing significant in the storytelling of Multiverse of Madness, and Marvels new fashion of introducing cameos to bring more audiences to watch these movies has become quite unremarkable.

在這兩個例子中，SHAP 和 LIME 都能正確的判斷出正確的影評，但 SHAP 更能理解句子，LIME 只有找出幾個關鍵字來評斷，其中我認為 Madness 很明顯是負面的字詞，但 LIME 卻認為是 positive。

總結來說我覺得在短句方面兩中方法都表現得差不多，而在長句是 SHAP 表現比較好，我想應該是 LIME 在分析時是基於局部線性近似，而造成無法看清整個句子所導致的。

4.Try 3 different input sentences for attacks

4.1Synonym Replacement

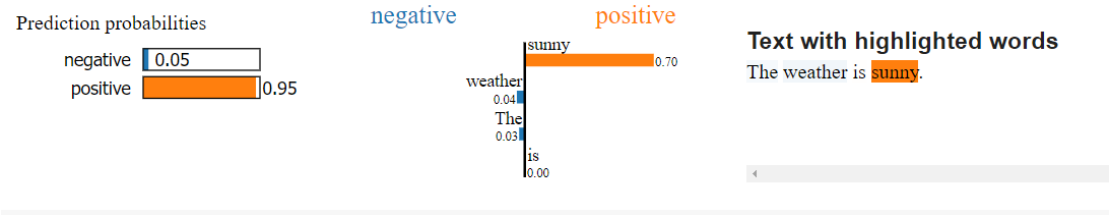
1. The weather is sunny.

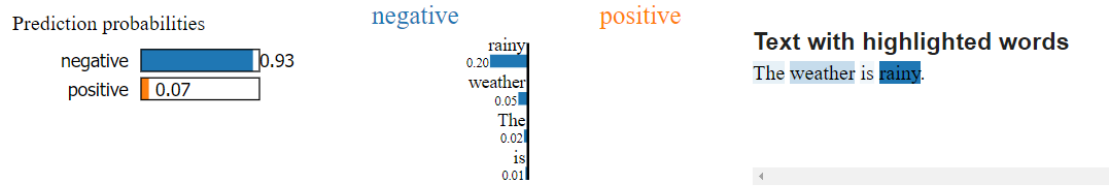
2. The weather is rainy.

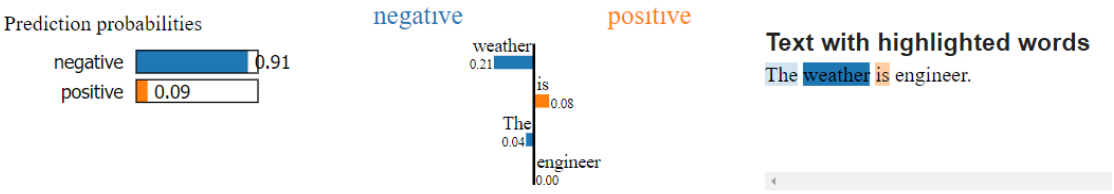
3. The weather is engineer.

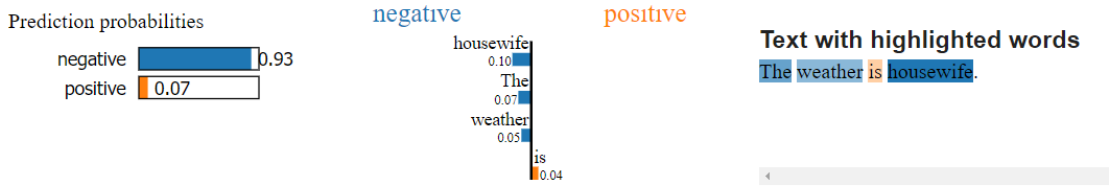
4. The weather is housewife.

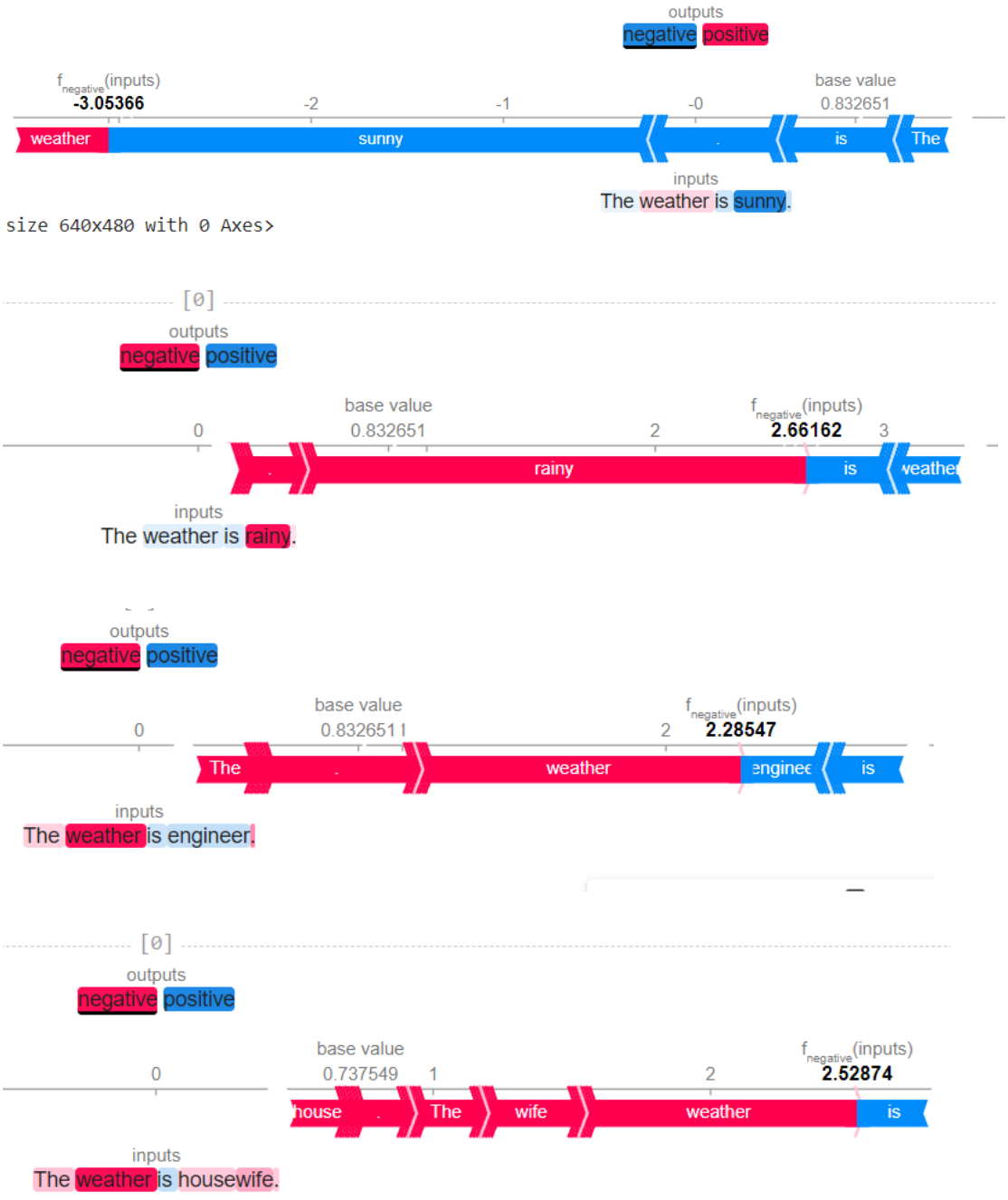
藉由單字的替換，會大大影響結果，接上特殊名詞它就完全搞不懂句子的意思









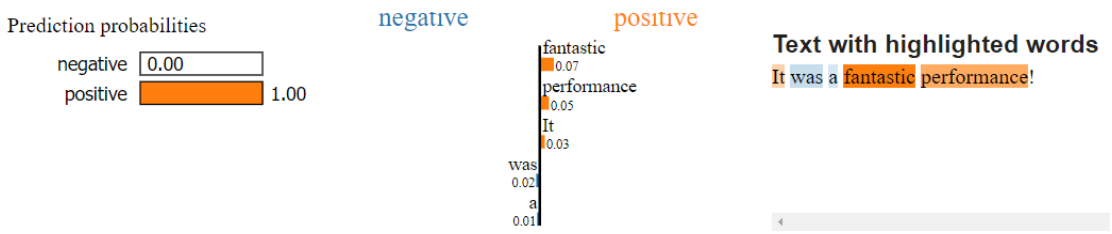


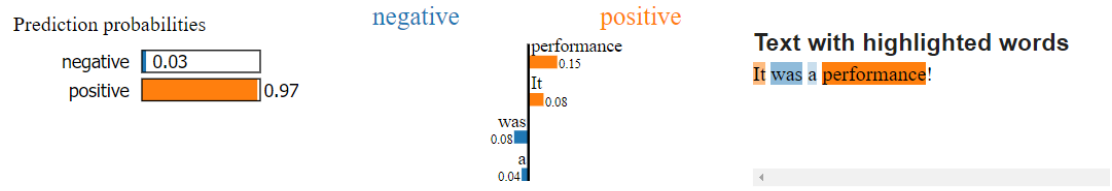
4.2 Word Deletion

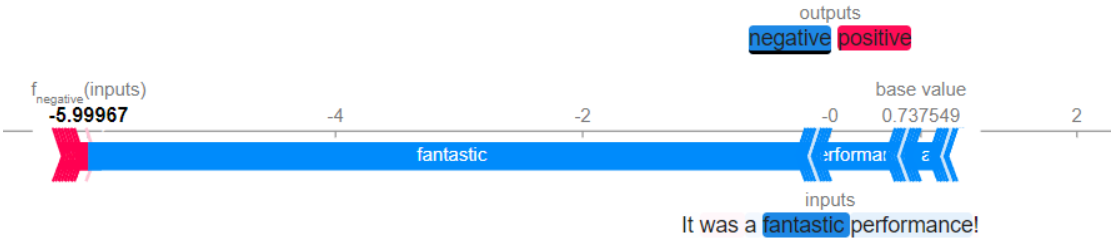
1. It was a fantastic performance!

2. It was a performance!

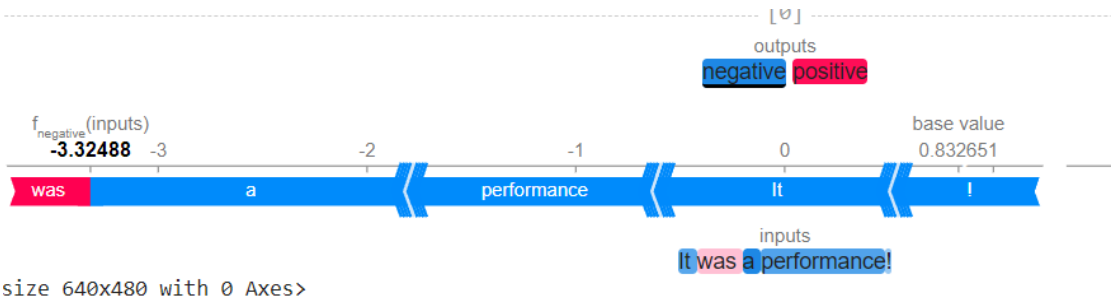
刪除掉句中的形容詞，但此範例沒有造成太大的差別








```
gure size 640x480 with 0 Axes>
```



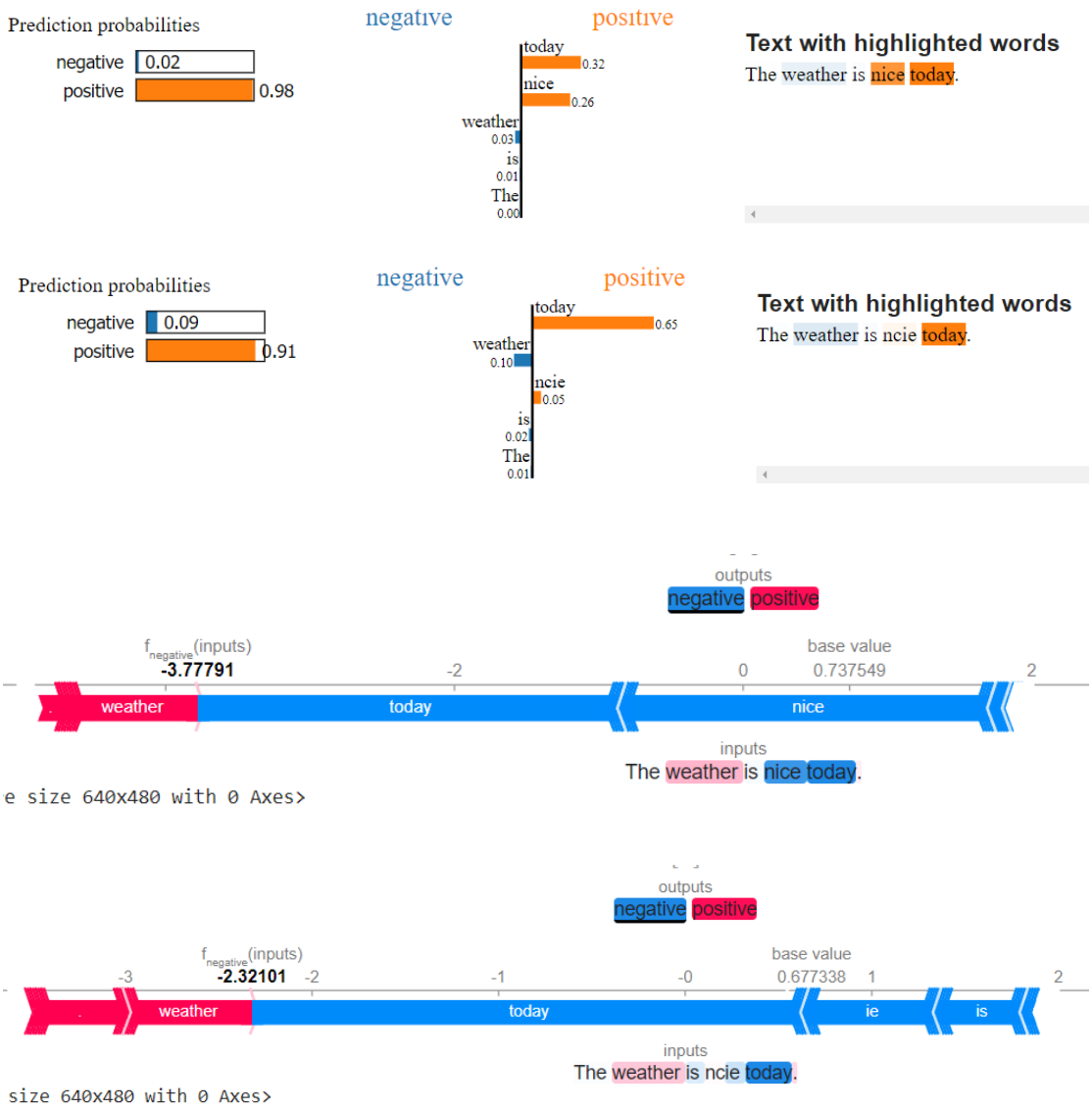

```
size 640x480 with 0 Axes>
```

4.3 Character Level Transformation

1. The weather is nice today.

2. The weather is ncie today.(改變 nice)

讓句中產生拼字錯誤，造成模型不懂，發現模型會認不出句子，造成確認正負時不能完全確定



4.4 如何防止攻擊

1.數據增強：在訓練數據中加入同義詞替換、詞刪除和字元級變換等擾動數據，使模型學習這些擾動的影響。

2.對抗訓練：產生對抗樣本並將其納入訓練集中，以增強模型的穩健性。

3.正規化技術：使用 Dropout 和權重懲罰等正規化技術，減少模型對特定輸入模式的過度擬合。

4.檢測機制：在預測階段加入檢測機制，以識別並過濾潛在的對抗樣本。

5. Describe problems you meet

一開始跑.ipynb 時發現無法完整跑完，有卡了幾天才看到,Microsoft Team 有人也有相同的問題，最後才解決。

雖然我在此作業形容 SHMP 和 LIME 的做法時很簡短，但在理解做法時花了很多時間，有自行上網查解釋

SHAP: https://medium.com/sherry-ai/xai-%E9%80%8F%E9%81%8E-lime-%E8%A7%A3%E9%87%8B%E8%A4%87%E9%9B%9C%E9%9B%A3%E6%87%82%E7%9A%84%E6%A8%A1%E5%9E%8B-23898753bea5

LIME: https://medium.com/sherry-ai/xai-%E9%80%8F%E9%81%8E-lime-%E8%A7%A3%E9%87%8B%E8%A4%87%E9%9B%9C%E9%9B%A3%E6%87%82%E7%9A%84%E6%A8%A1%E5%9E%8B-23898753bea5

不過這次作業還挺有趣的，尤其是在 attacks，嘗試了很多。