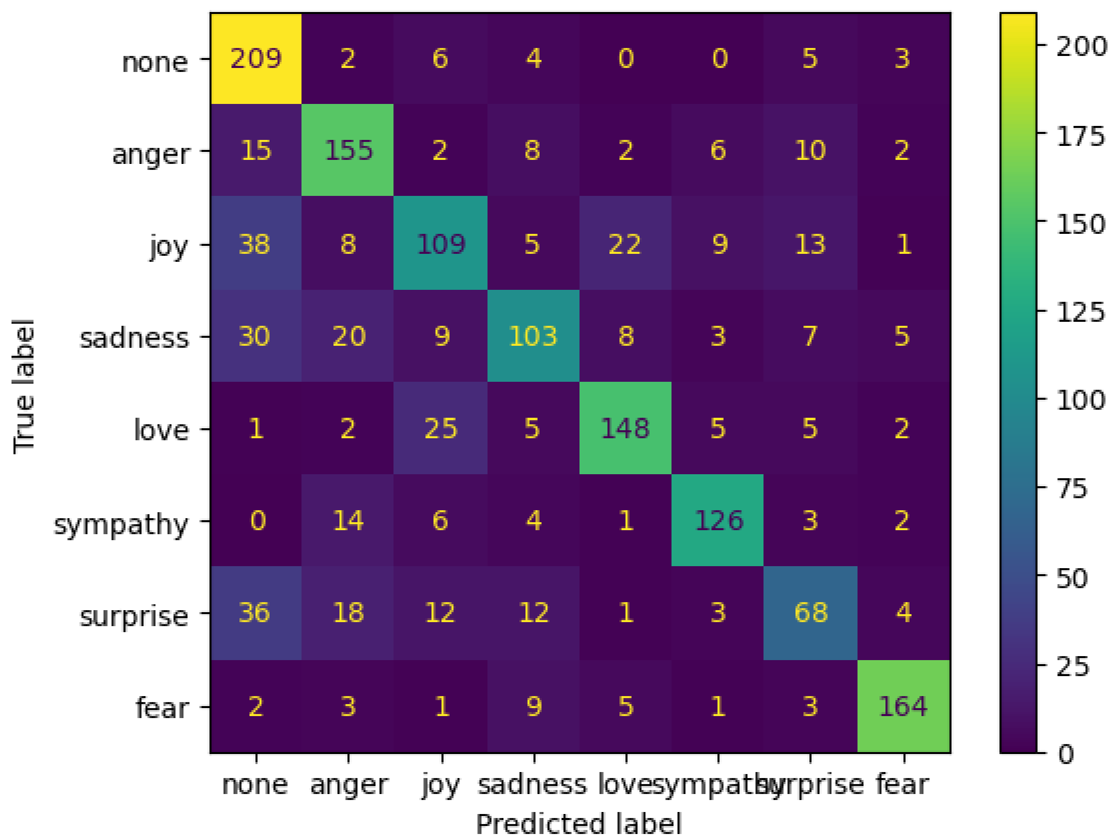
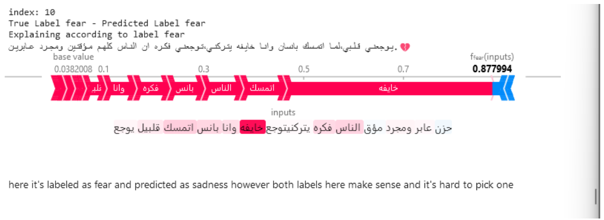
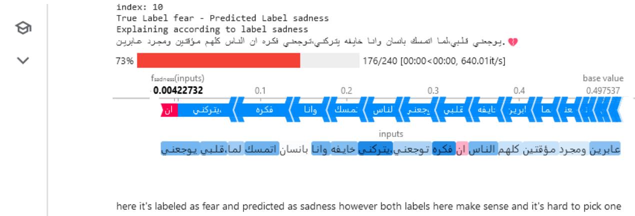
We will be using the light stemming model from now.

	precision	recall	f1-score	support
none	0.63	0.91	0.75	229
anger	0.70	0.78	0.73	200
joy	0.64	0.53	0.58	205
sadness	0.69	0.56	0.61	185
love	0.79	0.77	0.78	193
sympathy	0.82	0.81	0.82	156
surprise	0.60	0.44	0.51	154
fear	0.90	0.87	0.88	188
accuracy			0.72	1510
macro avg	0.72	0.71	0.71	1510
weighted avg	0.72	0.72	0.71	1510

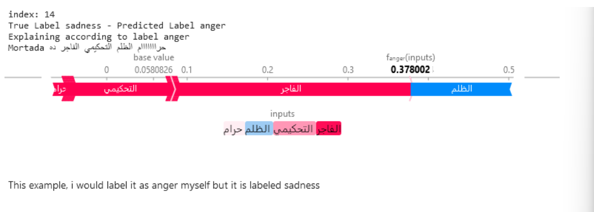


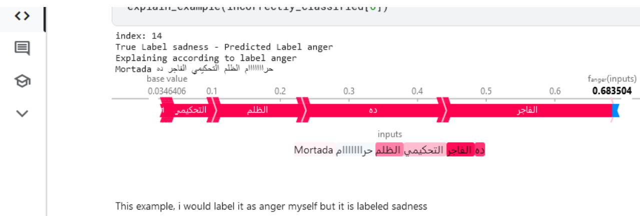
Light Stemming vs Raw Data:



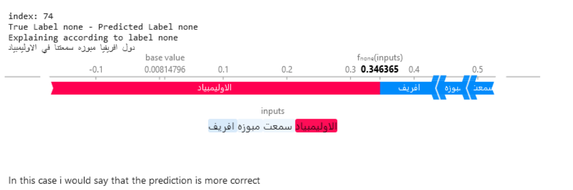


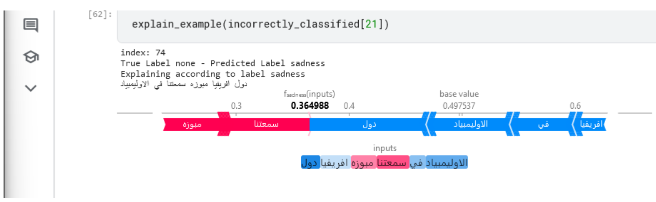
Here the model made the correct prediction with light stemming.



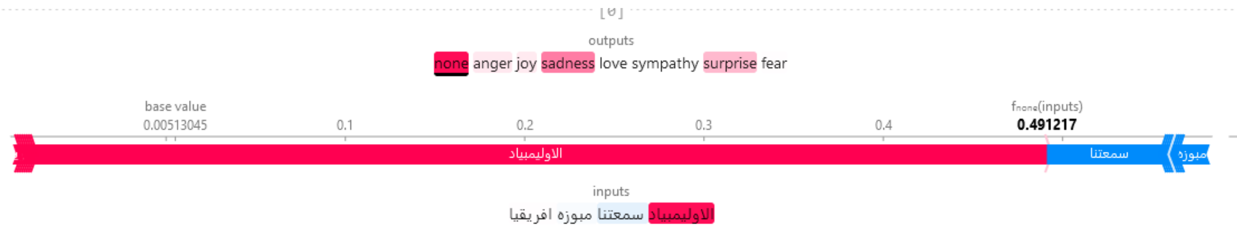


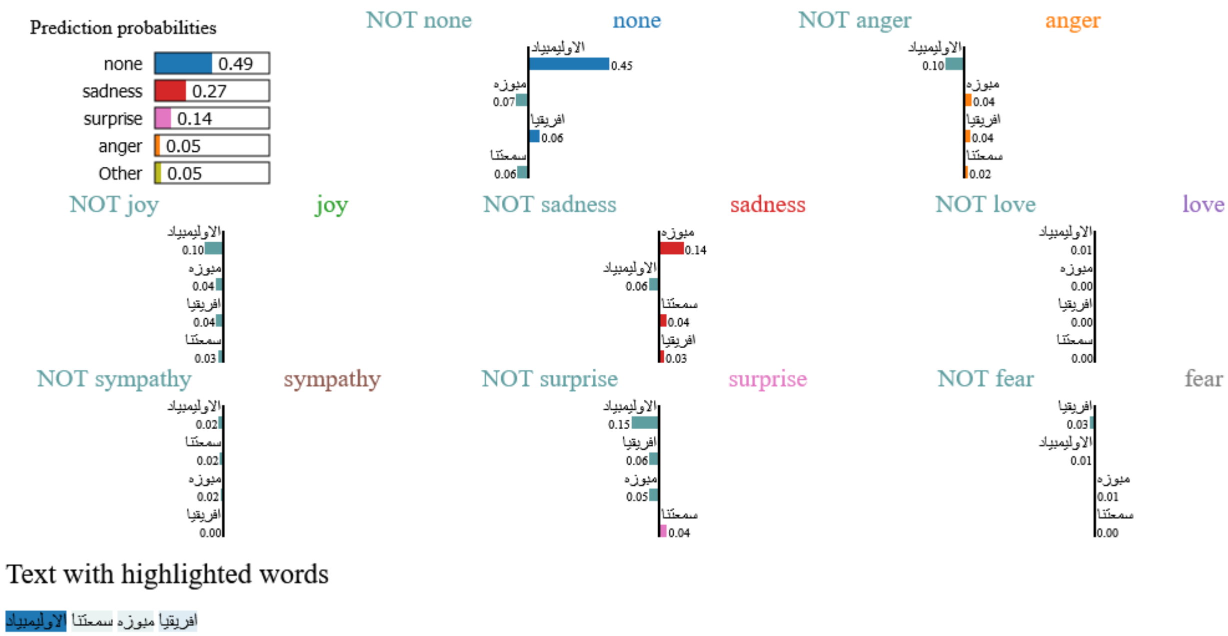
The model stopped using stopwords as explanations.





This was a case which I agreed with the prediction more than the label, but now the model got the label correctly. The explanation is bad too. ***(try the correct word)

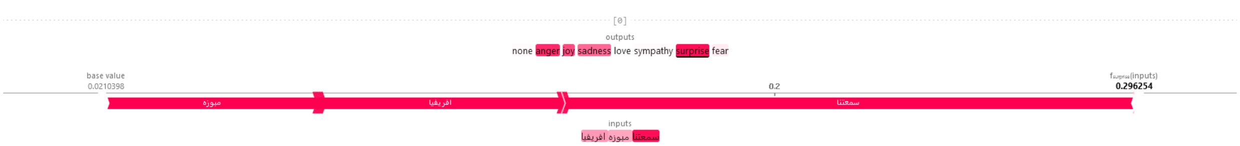




No significant change after correction?

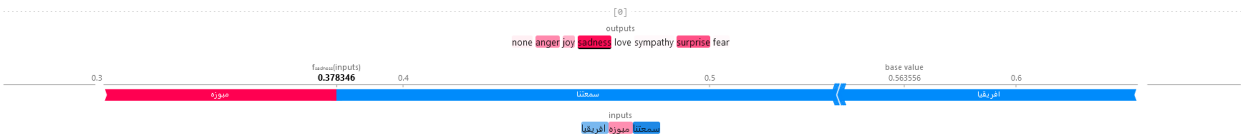
Only that the word turned from being not-none to none but its score is not significant enough to change the outcome in either cases

Let's us try without the problematic word:



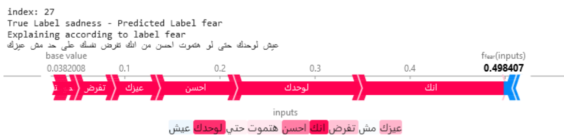

```
explanation = explainer.explain_instance(df_test[stemtype].iloc[testi], model_predict, num_features=10, labels = range(8))

  # showing the explanation
explanation.show_in_notebook()
```

Now let us fix the word:

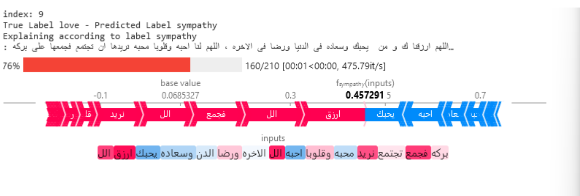


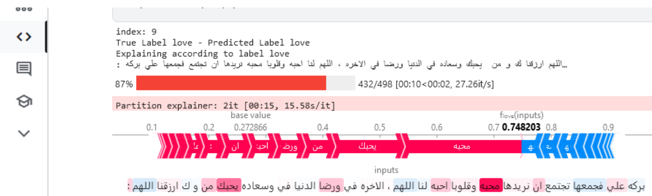

```
explanation = explainer.explain_instance(df_test[stemtype].iloc[testi], model_predict, num_features=10, labels = range(8))

  # showing the explanation
explanation.show_in_notebook()
```

We returned to sadness once more.



This one just got more confusing.





The model got confused in this case, as light stemming loses some of the context, so the mistake is understandable.

Conclusion:

The explanations got better but some of the context was lost.

Shap vs Lime

Interruption:

Some errors were found in light stemming during the comparison of lime and shap.

Raw data:

💔.يوجعني قلبي،لما اتمسك بانسان وانا خايفه يتركني،توجعني فكره ان الناس كلهم مؤقتين ومجرد عابرين

Light Stemming:

يوجع قلبيل اتمسك بانس وانا خايفه يتركنيتوجع فكره الناس مؤق ومجرد عابر حزن

While the main contributor to the label is the same, the other words are assigned different importance in different explainers.

The same words contribute to different labels according to different explainers.

Here, both have mostly the same explanation, but the words are assigned different weights.

Here, LIME gave better insight into the workings of the model by providing the probabilities. We can see that even the model understands that there are mixed feelings.

Observe the weight of each words in the following examples:

1)

2)

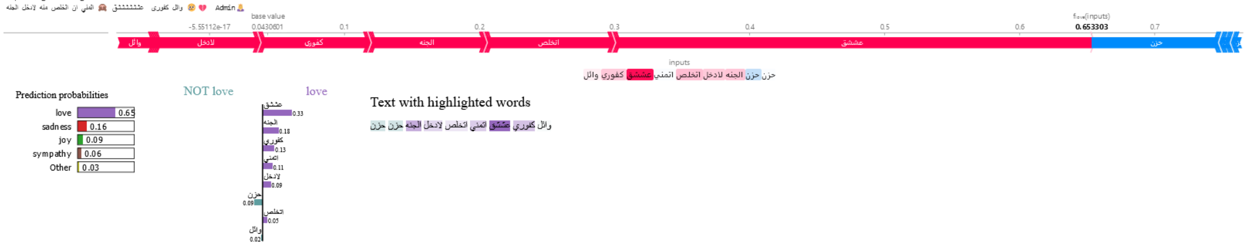
```
index: 14
True Label sadness - Predicted Label anger
Explaining according to label anger
Mortada ده الفاجر التحكيمي الظلم حرام!!!!!!
```


```
explain_example(testi)
```

```
index: 14
True Label sadness - Predicted Label anger
Explaining according to label anger
Mortada ده الفاجر التحكيمي الظلم حرام!!!!!!
```

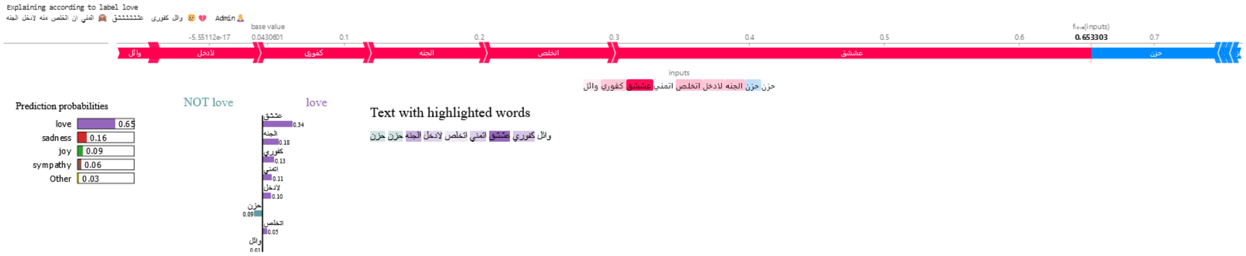
3)

```
index: 15
True Label love - Predicted Label love
Explaining according to label love
Admin 🙆 💔 😢 وائل كفوري عشششششششق 🙈 اتمني ان اتخلص منه لادخل الجنه
```




```
explain_example(testi)
```

```
index: 15
True Label love - Predicted Label love
Explaining according to label love
Admin 🙆 💔 😢 وائل كفوري عشششششششق 🙈 اتمني ان اتخلص منه لادخل الجنه
```



4)

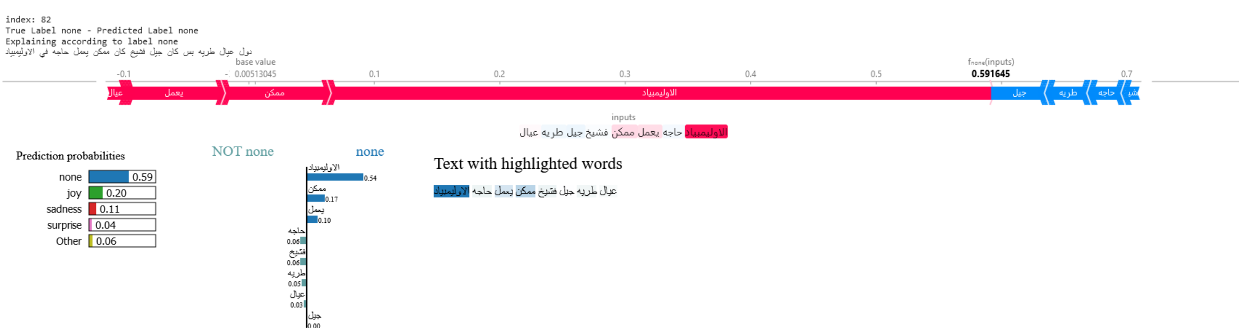
Due to the nature of its calculations, SHAP is very consistent in its output. Lime on the other hand is inconsistent due to its random nature.

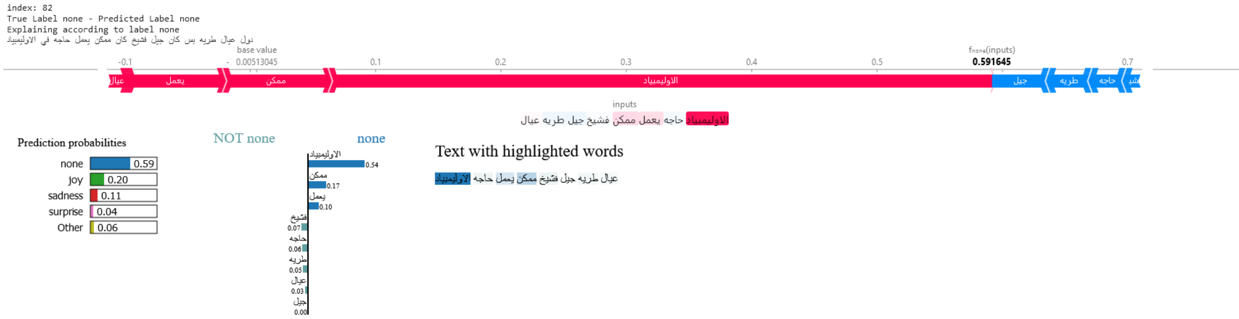
Explaining the same example using Lime multiple times gives different weights each time but it’s usually generally correct.

The most significant word is almost always the same and the probabilities don’t appear to change but it would make a difference if the values were close to each other such that a 0.01 is enough to make a difference.

Here the order of significance is different.

Here we can observe that some words appeared in the first sentence but not the second sentence and vice versa.





As we can see here, a small change was enough to make a difference.

The last word switched from None to Not-none.

Now, we attempted to figure out why some words don't have weight in shap but have high weight in lime:

Let's return to this example once more.

The word 'معقول' has a weight of -0.09 in lime (it was verified by multiple runs) but appears to have no significant weight with shape (-0.007)

Across multiple runs I tried to remove words that has little to no weight in both lime and shap as following:

As we can observe in lime, other than the removed word, the words have the same order and a little difference in weight.

But in Shap, each removed word has a significant effect on the rest of the weights even if the removed word’s weight is not significant itself.

Conclusion:

Due to the way of calculation, Lime has little in the way of stability when running the same example multiple times but is surprisingly robust when some of the input is changed or dropped.

Shap on the other hand is very consistent in its calculations when running the same example, but it is greatly affected by small changes in the input and seemingly unrelated words may have significant changes in the output.

Interruption (again):

Some errors were found in light stemming during the comparison of lime and shap.

Raw data:

دول افريفيا مبوزه سمعتنا في الاوليمبياد

Light Stemming:

افريف مبوزه سمعت الاوليمبياد

There is a spelling mistake here, and some removed stopwords that need to be looked into.

And another one:

Raw data:

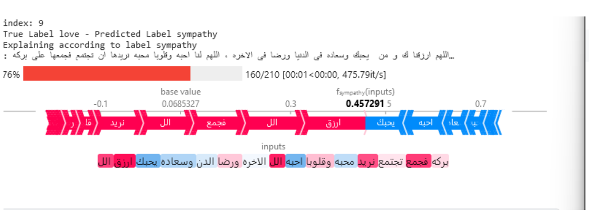
محدش عارف هما تعبوا اد ايه عشان يوصلوا الاوليمبياد و هم مش ناقصين تعليقات الجهله في اللعبه لادائهم و لا تعليقات المصريين عموما علي لبسهم

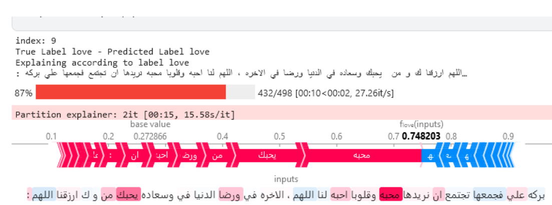
Light Stemming:

محدش عارف تعب اد عشان يوصل الاوليمبياد مش ناقص تعليق الجهله اللعبه لادائ تعليق المصر عمو لبس

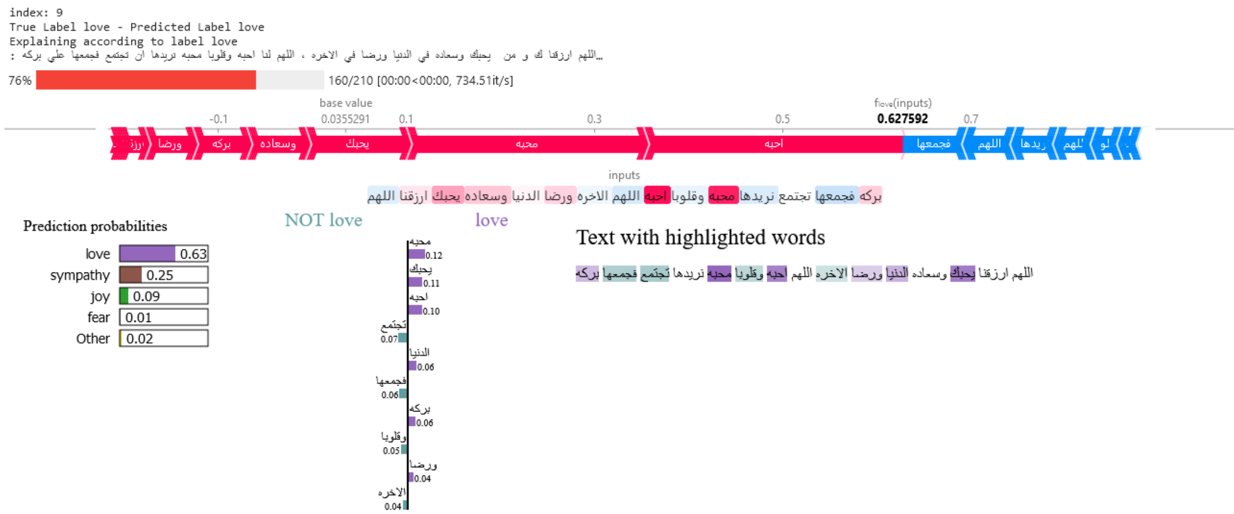
There are some removed stopwords that need to be looked into.

Using Cleaned data without stemming:





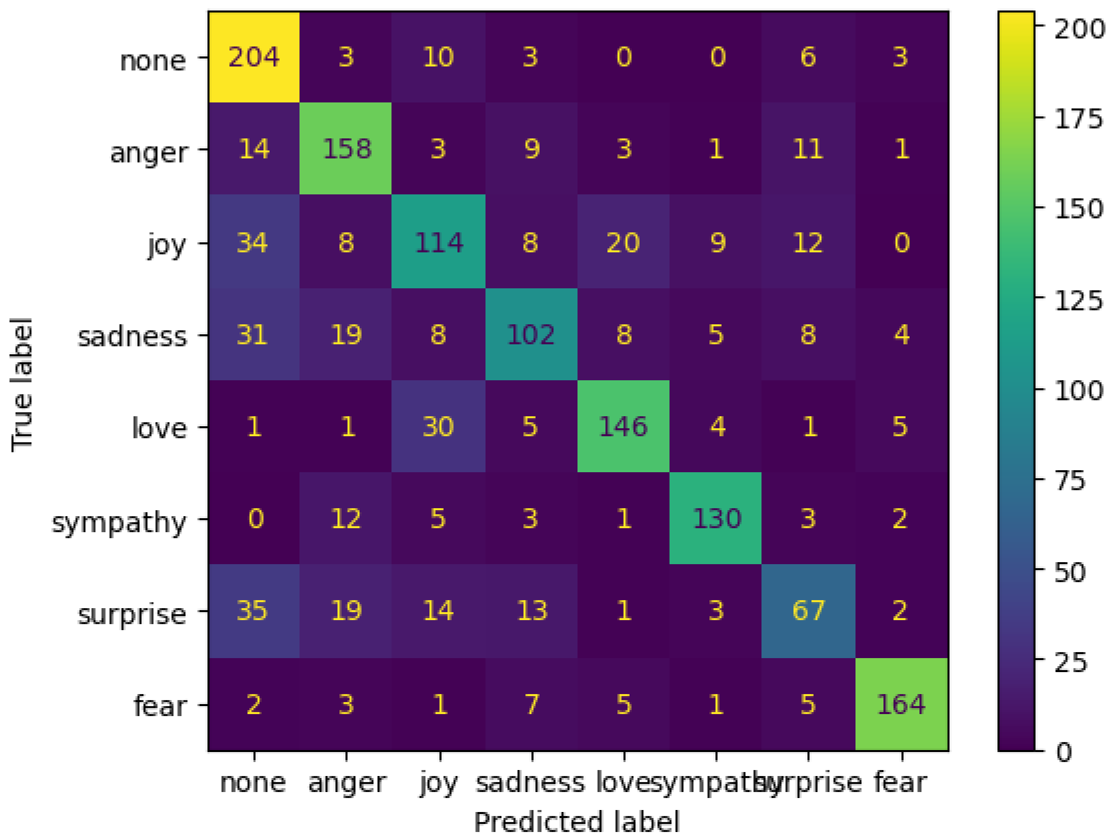
There was this error that I attributed to a loss of context but without stemming, the model got it correctly once more:



So, context can be lost with stemming but it appears that without stemming, it may get better.

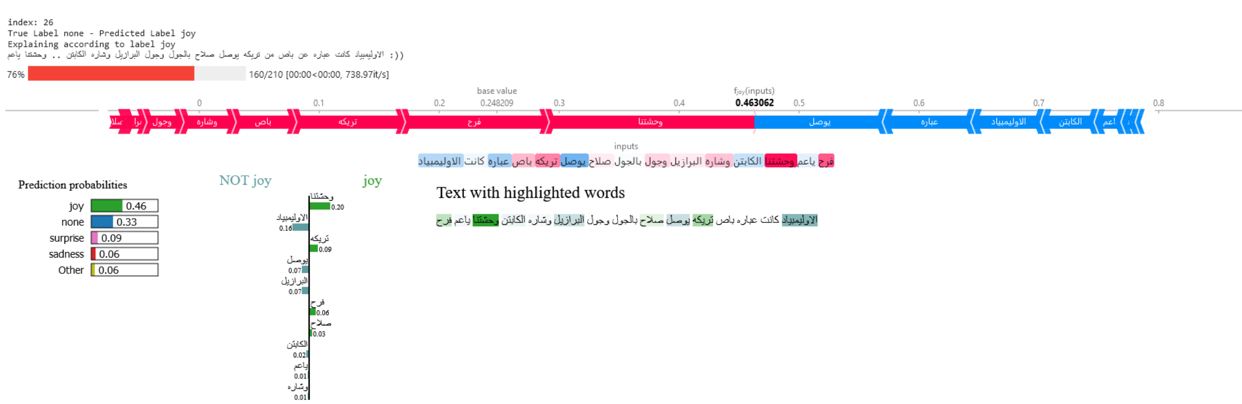
Since we will be using cleaned data now, here are the metrics:

	precision	recall	f1-score	support
none	0.64	0.89	0.74	229
anger	0.71	0.79	0.75	200
joy	0.62	0.56	0.58	205
sadness	0.68	0.55	0.61	185
love	0.79	0.76	0.77	193
sympathy	0.85	0.83	0.84	156
surprise	0.59	0.44	0.50	154
fear	0.91	0.87	0.89	188
accuracy			0.72	1510
macro avg	0.72	0.71	0.71	1510
weighted avg	0.72	0.72	0.71	1510



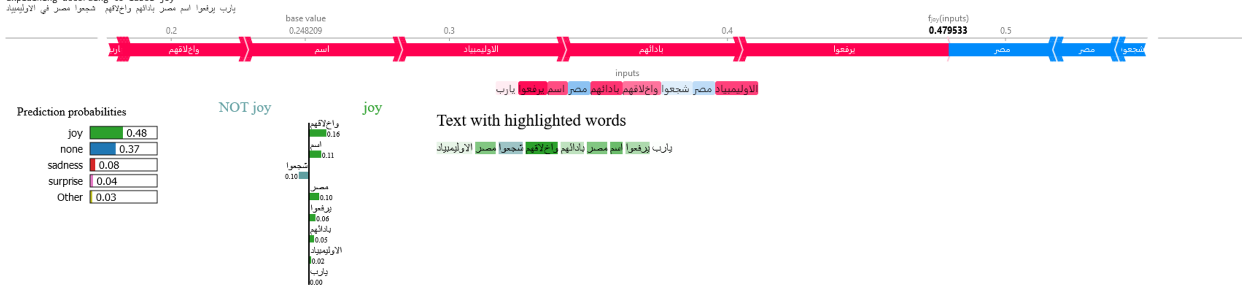
By comparing the confusion matrix with the previous one, we can observe that the box that indicates (true = none, pred = joy) increased from 6 to 10.

That was despite the fact that context appears to be more complete without stemming so lets examine those 10 samples.

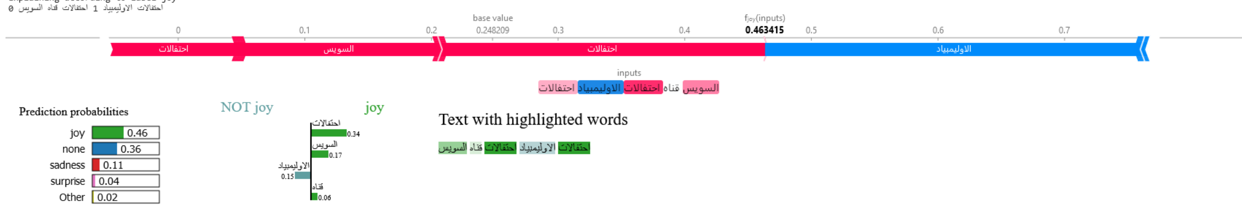


index: 100
True Label none - Predicted Label joy
Explaining according to label joy
شنو يعجبك بالكويت ماعمري زرتها لكن يعجبني احمد السيار

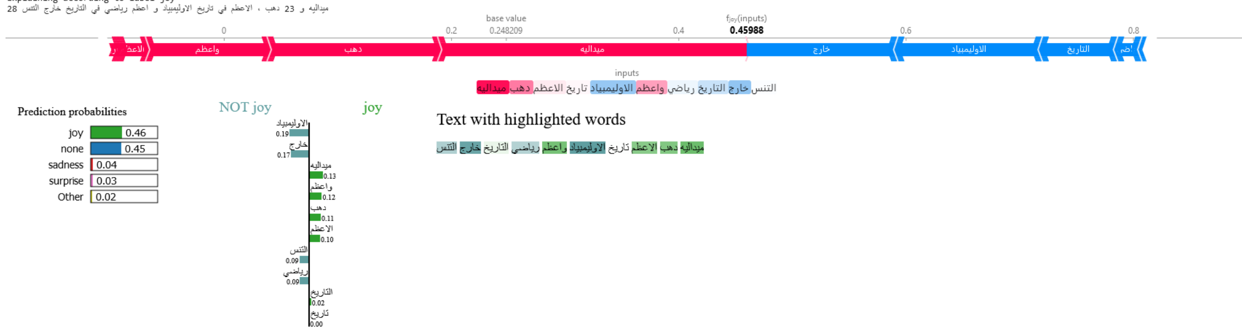
index: 325
True Label none - Predicted Label joy
Explaining according to label joy
يارب يرفعوا اسم مصر باداءهم واخلاقهم شجعوا مصر في الاوليمبياد



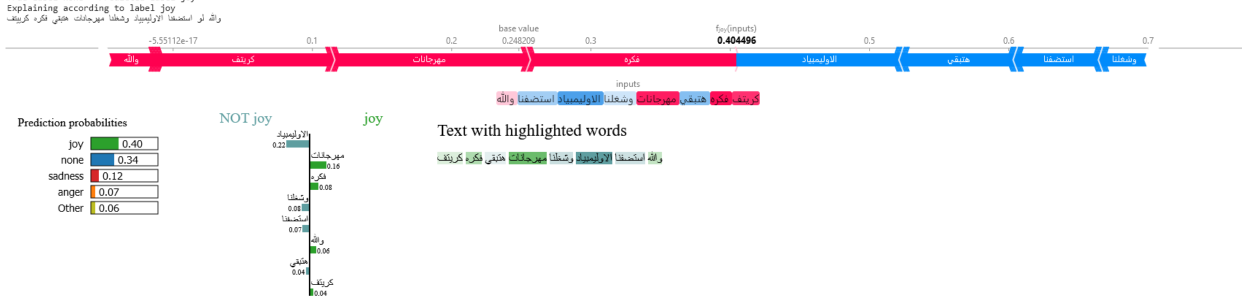
index: 801
True Label none - Predicted Label joy
Explaining according to label joy
0 احتفالات الاوليمبياد 1 احتفالات قناه السويس

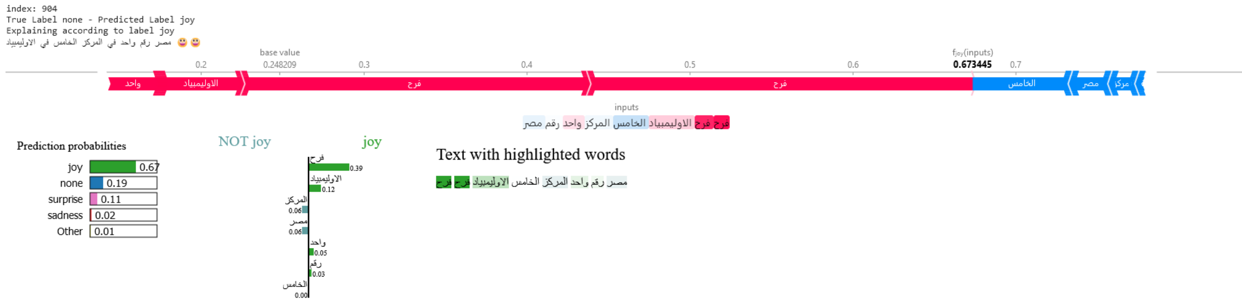


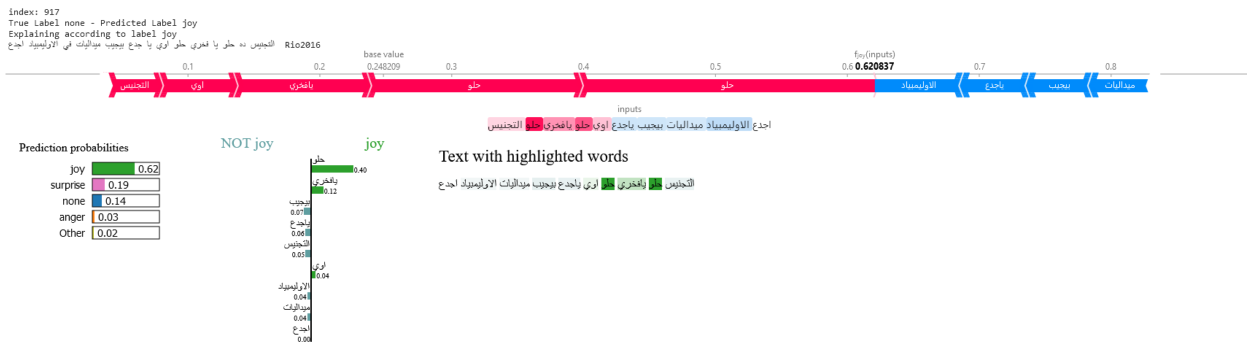
index: 815
True Label none - Predicted Label joy
Explaining according to label joy
28 ميداليه و 23 دهب ، الاعظم في تاريخ الاوليمبياد و اعظم رياضي في التاريخ خارج التنس

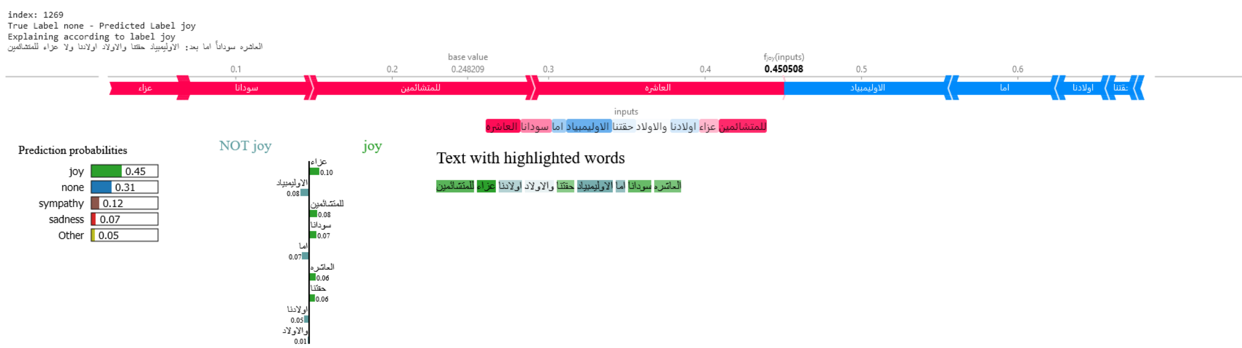


index: 874
True Label none - Predicted Label joy
Explaining according to label joy
والله لو استضفنا الاوليمبياد وشغلنا مهرجانات هتبقي فكره كرييتف







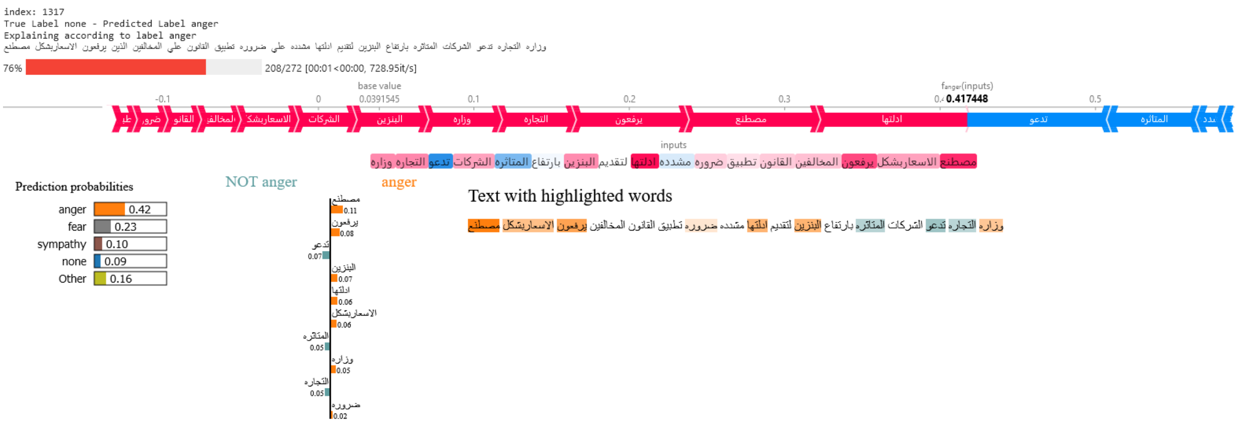


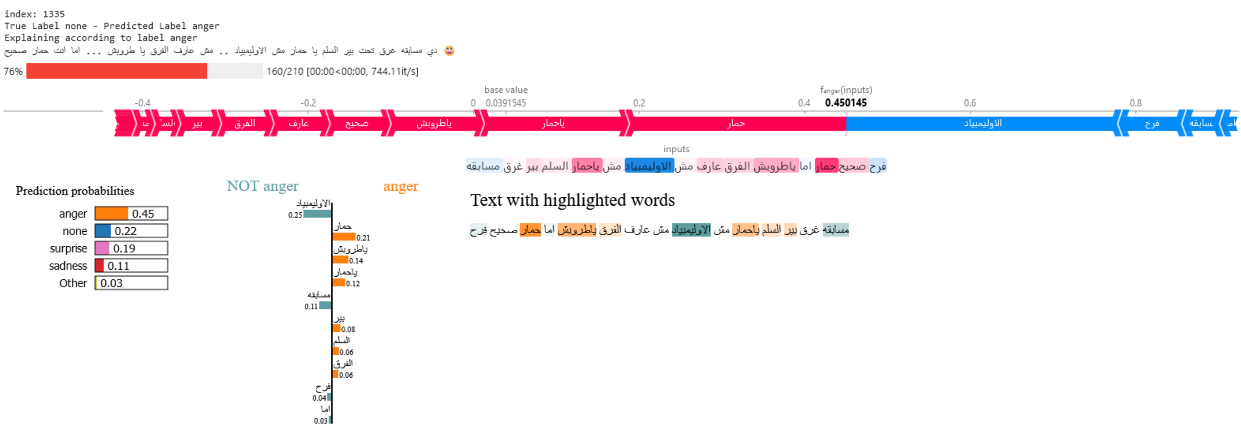
Some examples are understandable why the model predicted them as joy, while others are just confusing to me and the whole context did not help either.

The just context made the model more confused.

So, let's examine something a bit less confusing.

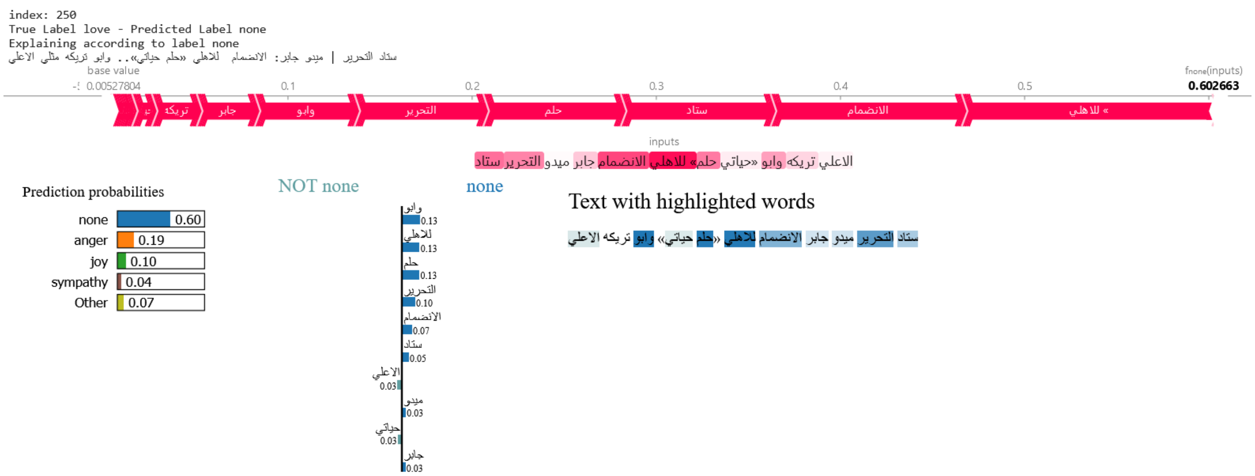
(true = none, pred = anger, total error = 3)





The first and last example are understandable why the model would label them as anger, and I am inclined to agree with its explanation. Especially the last one, as 3 curse words in one sentence is a convincing argument for why it is predicted anger.

(true = love, pred = none, total error = 1)



Most of the love samples in the dataset are the romantic kind of love, so the problem here is a lack of data in this very specific area.

(true = love, pred = joy, total error = 30)

index: 203

True Label love - Predicted Label joy

Original: احلام العنزي فخر الكويت الف الف مبروووووك بنت قبيلتي وبنت ديرتيييي??????????????????? كفو عيال الديره كل يوم اسم رافع اسمنا بشاره خير ان شاء الله

Cleaned: احلام العنزي فخر الكويت الف الف مبروك بنت قبيلتي وبنت ديرتي كفو عيال الديره يوم اسم رافع اسمنا بشاره خير شاء الله

I would say this one is mixed feelings.

index: 594

True Label love - Predicted Label joy

Original: : الدوله العثمانيه الاسلاميه ستبقى رغم انوف الاعداء ربي احفظ تركيا المسلمه من مكر الماكرين واحفظ كل بلاد المسلمين ♡

Cleaned: ♡ الدوله العثمانيه الاسلاميه ستبقى رغم انوف الاعداء ربي احفظ تركيا المسلمه مكر الماكرين واحفظ بلاد المسلمين

This one supports my earlier argument about romantic love and love for one’s country.

index: 823

True Label love - Predicted Label joy

Original: لون حياتك بحب وطاعه الله فهي طويله قصيره ... ايمان تقوه

Cleaned: لون حياتك بحب وطاعه الله طويله قصيره ايمان تقوه

Same as the above.

Conclusion on True = love:

It appears that most of them are predicted wrongly due to a few reasons:

1. The difference between romantic and non-romantic love
2. Mixed feelings
3. Misunderstood context
4. Generally confusing such as the following example:

index: 1430

True Label love - Predicted Label joy

Original: الهلال الساعه 7:30 بالتوفيق شباب الاتحاد &amp الاربعاء ديربي.. اتحاد الشرس

Cleaned: الاربعاء ديربي اتحاد الشرس الهلال الساعه بالتوفيق شباب الاتحاد

(true = surprise, pred = none, total error = 35)

All of them are due to “الاوليمبياد”

ALL OF THEM!!!