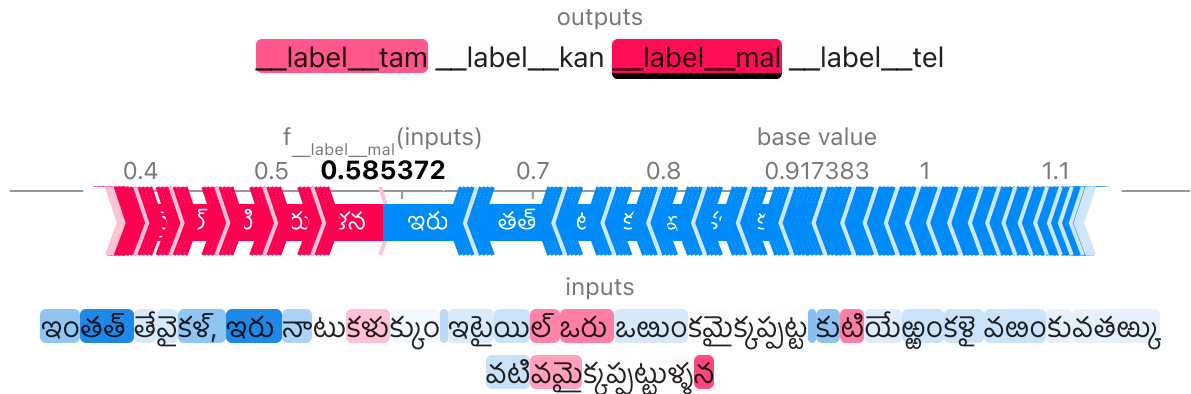


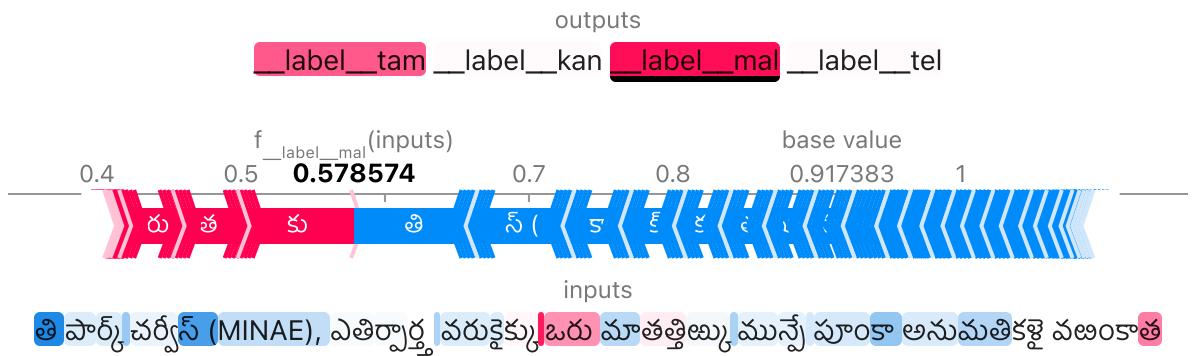
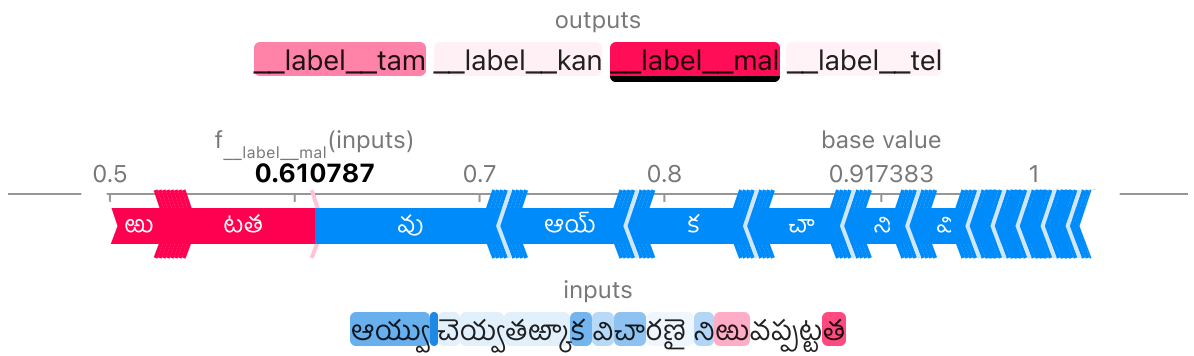
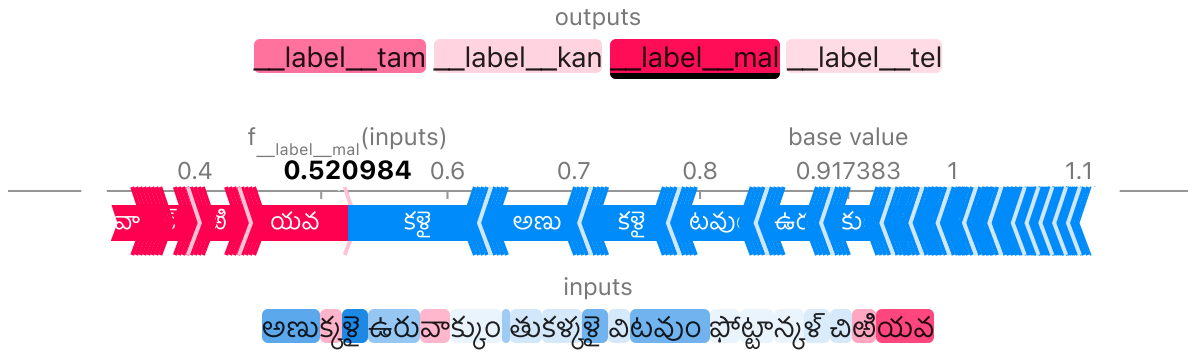
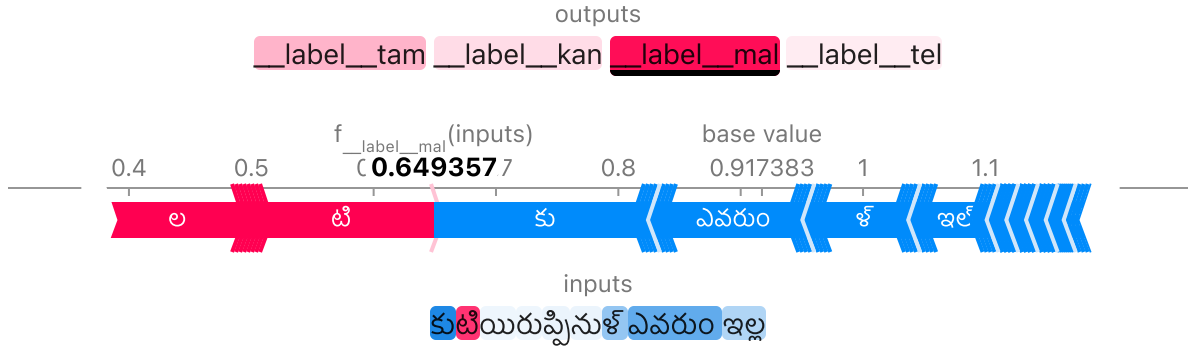
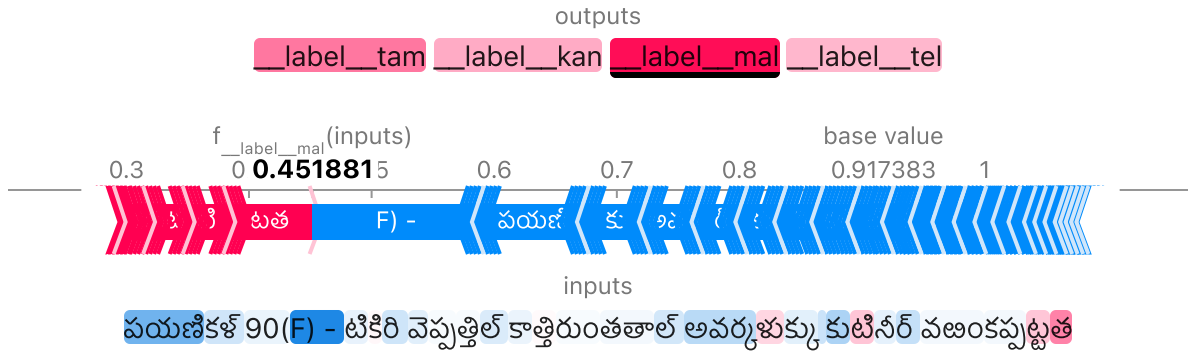
Character Level Explanations > 0.15

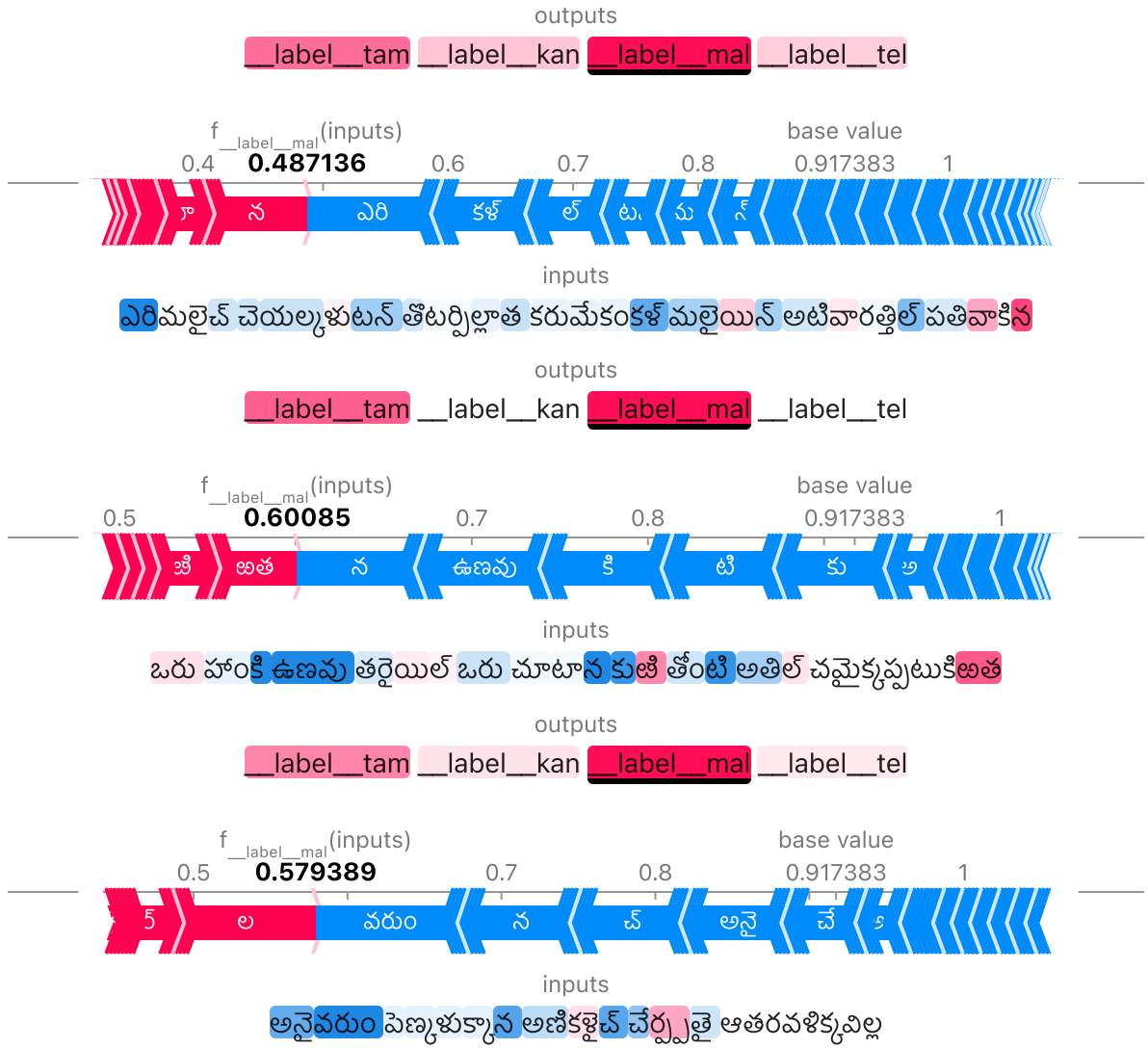
```
In [ ]: ix_array
```

```
Out[ ]: array([ 906, 1113, 1395, 1687, 2080, 2108, 2224, 2270, 2801])
```

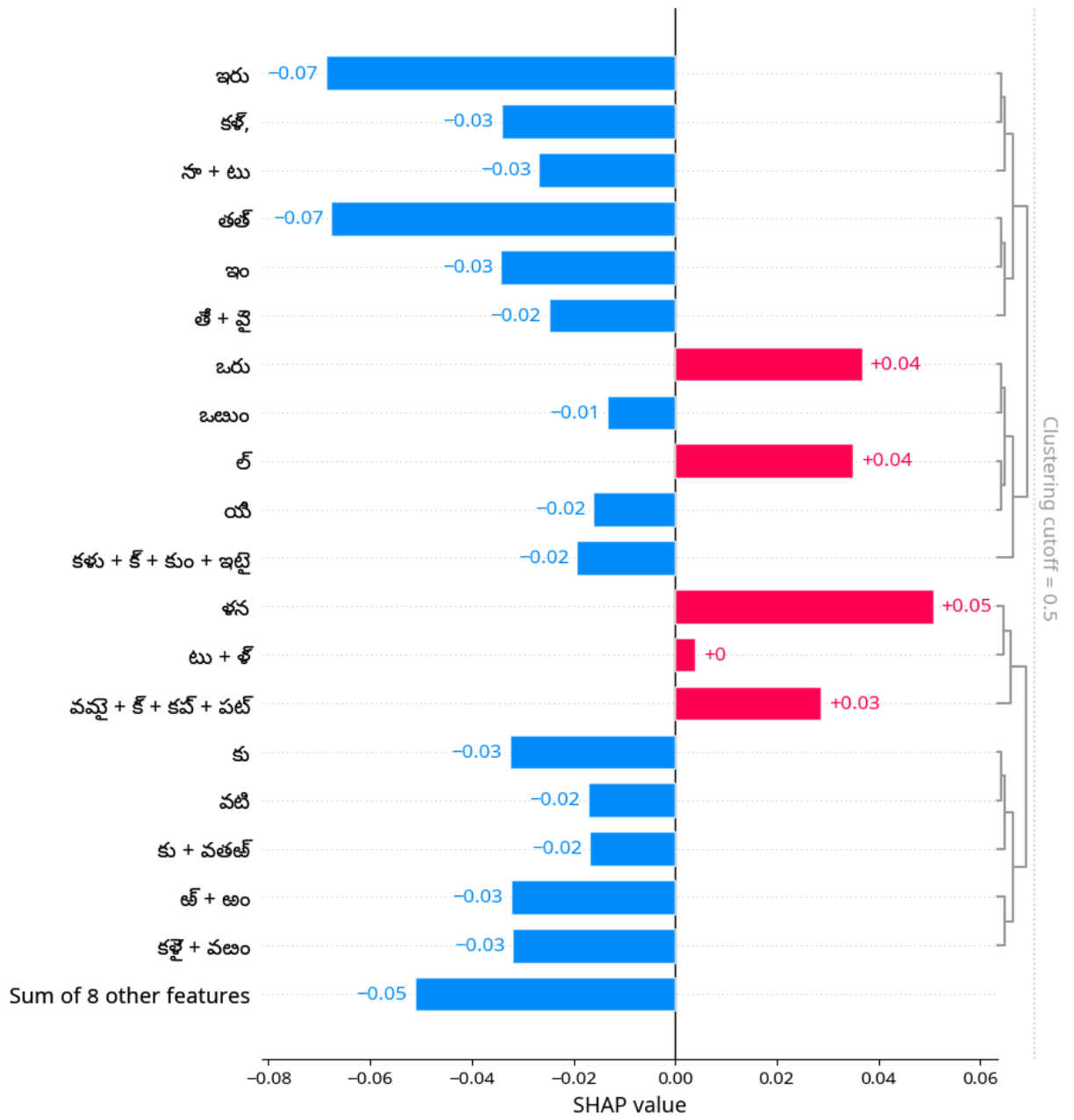
```
In [ ]: for i in ix_array:  
        shap.plots.text(shap_values[i])
```

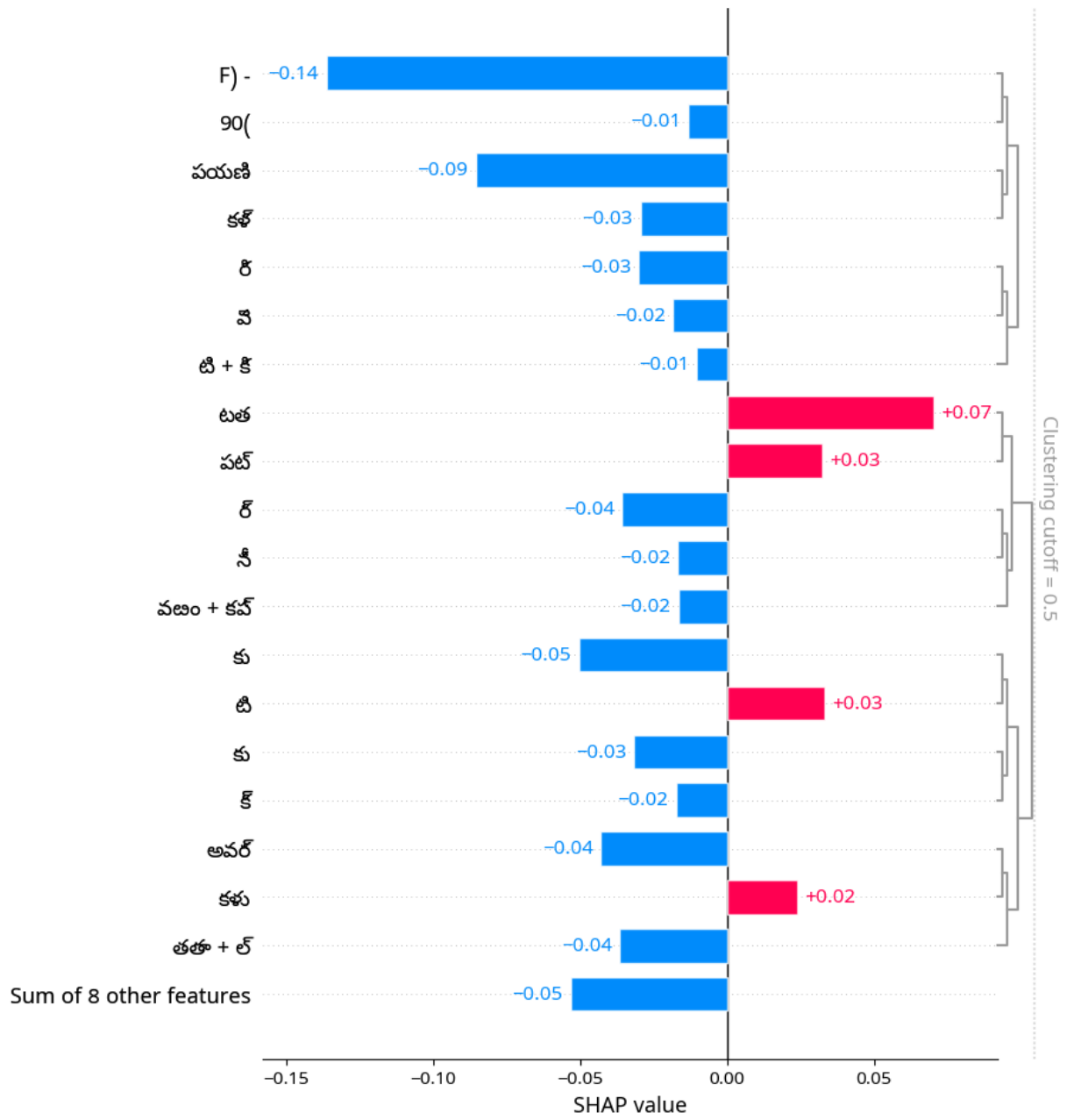


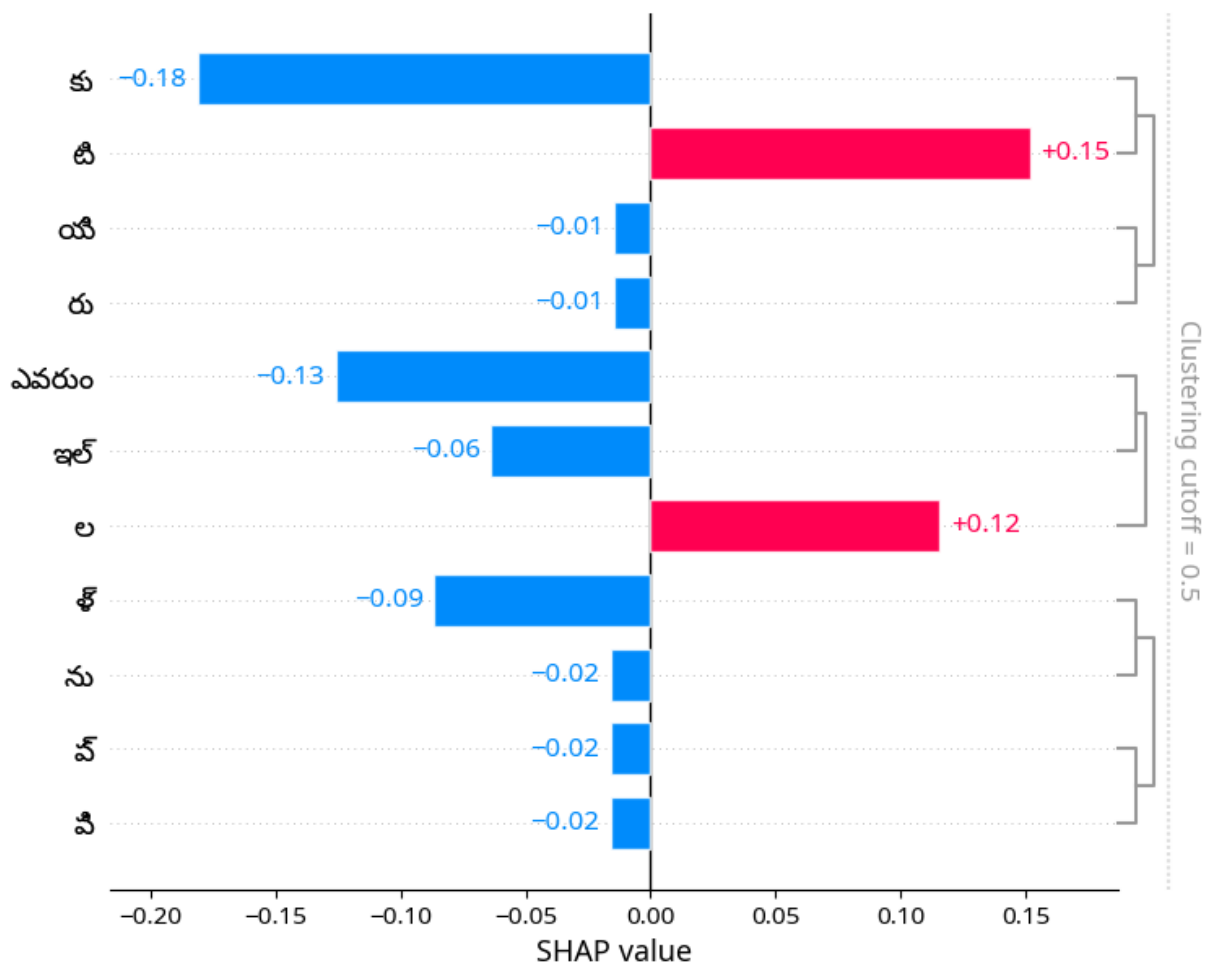


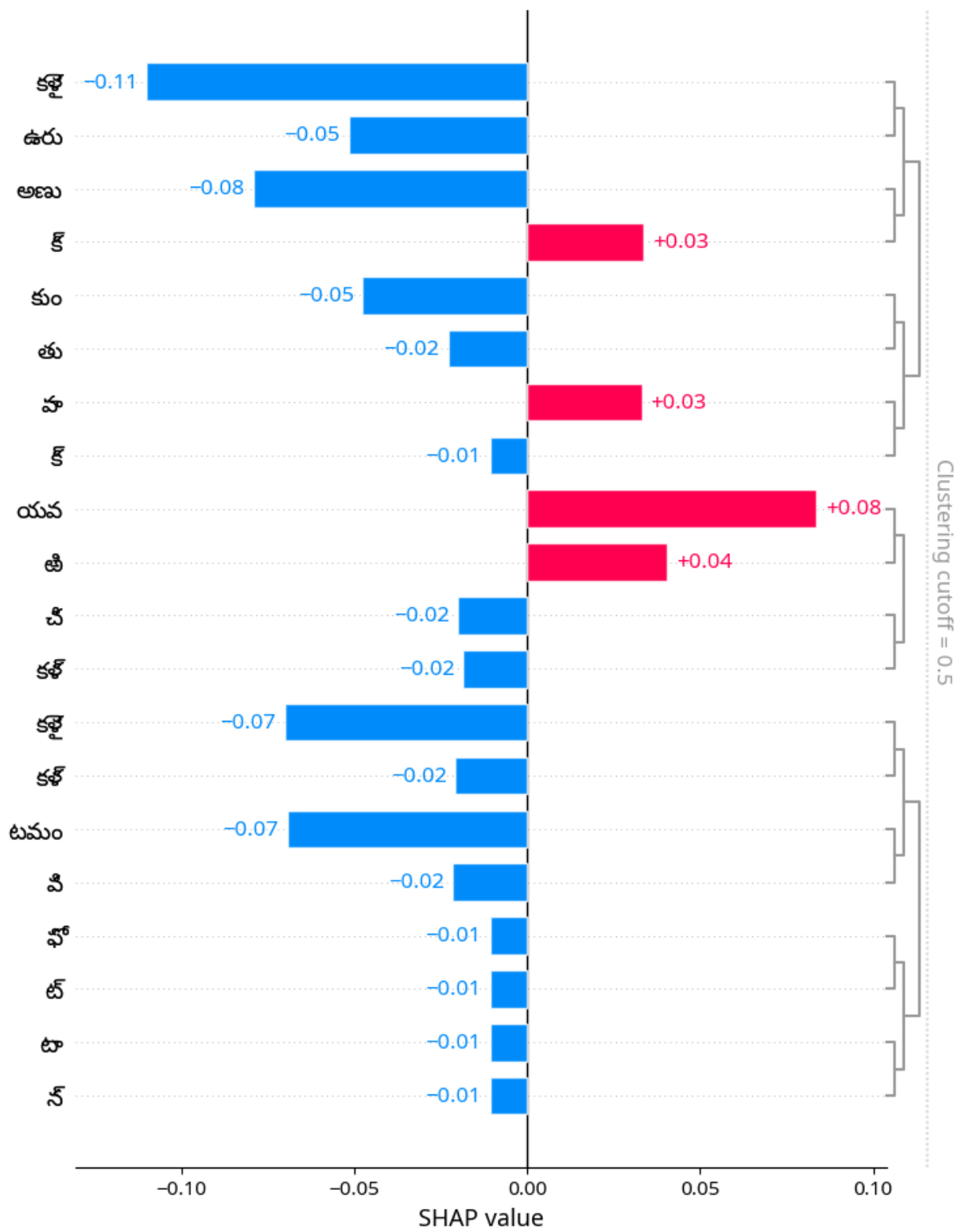


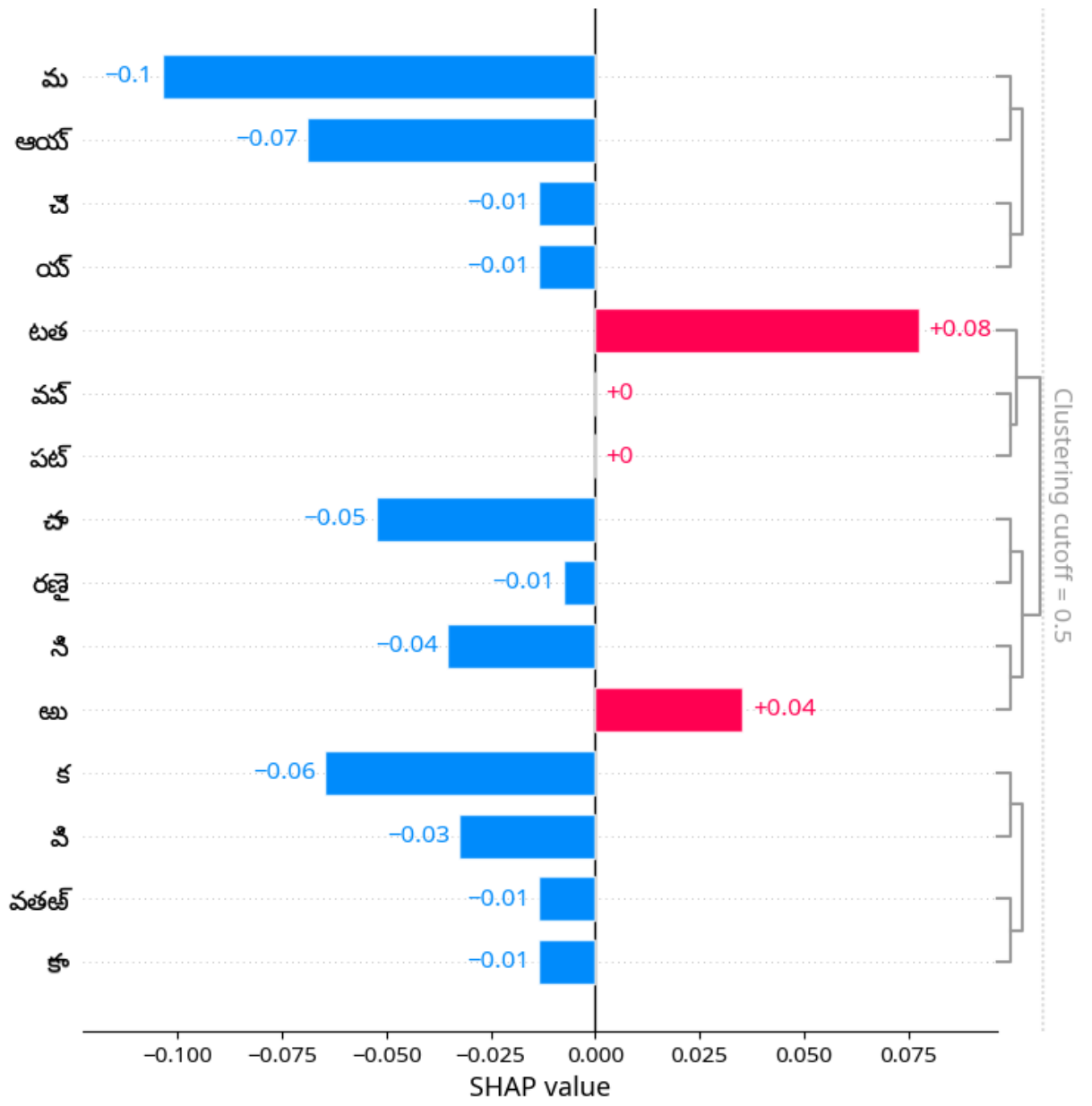
```
In [ ]: for i in ix_array:
        shap.plots.bar(shap_values[i][:,2], max_display=20)
```

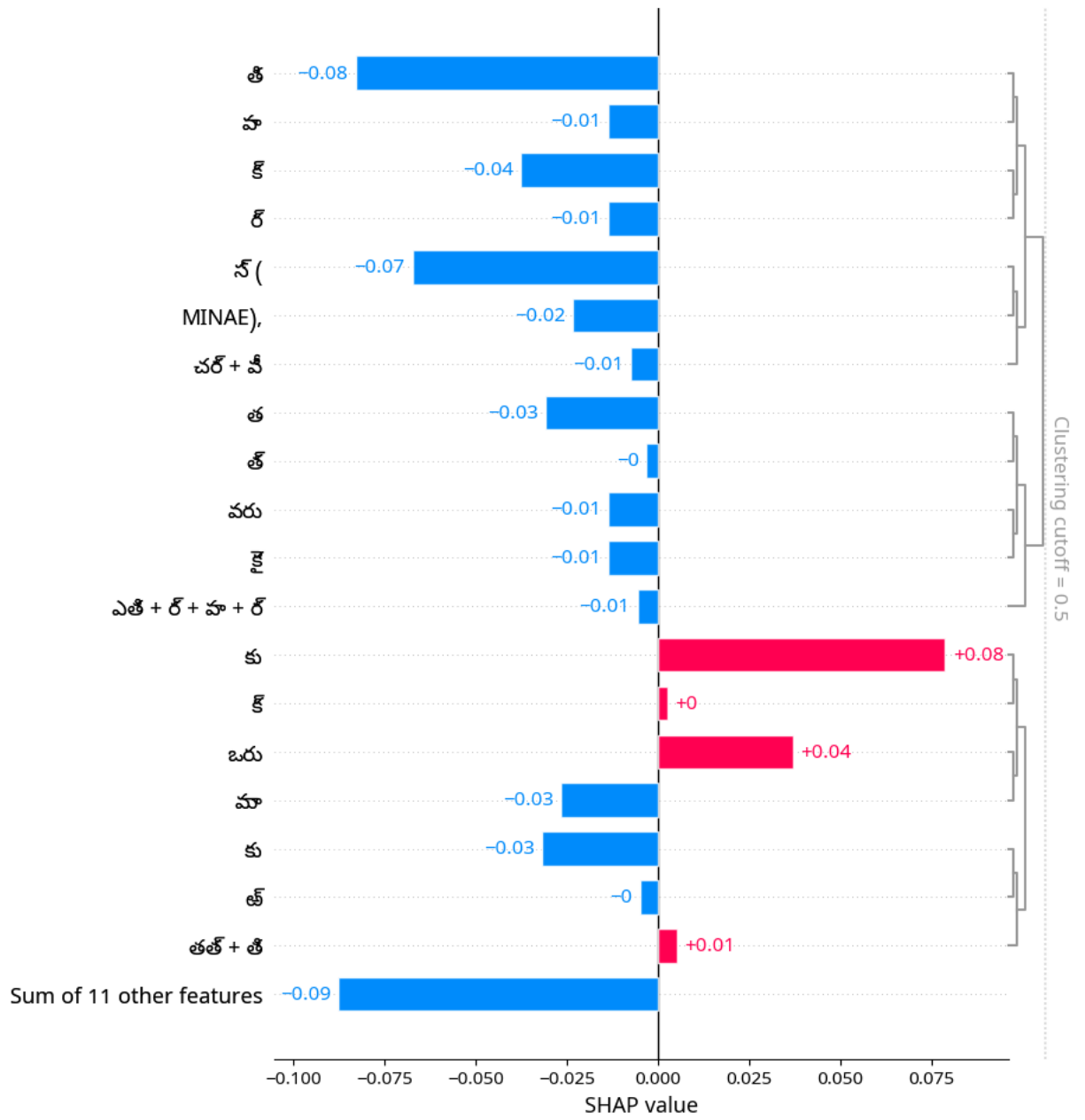


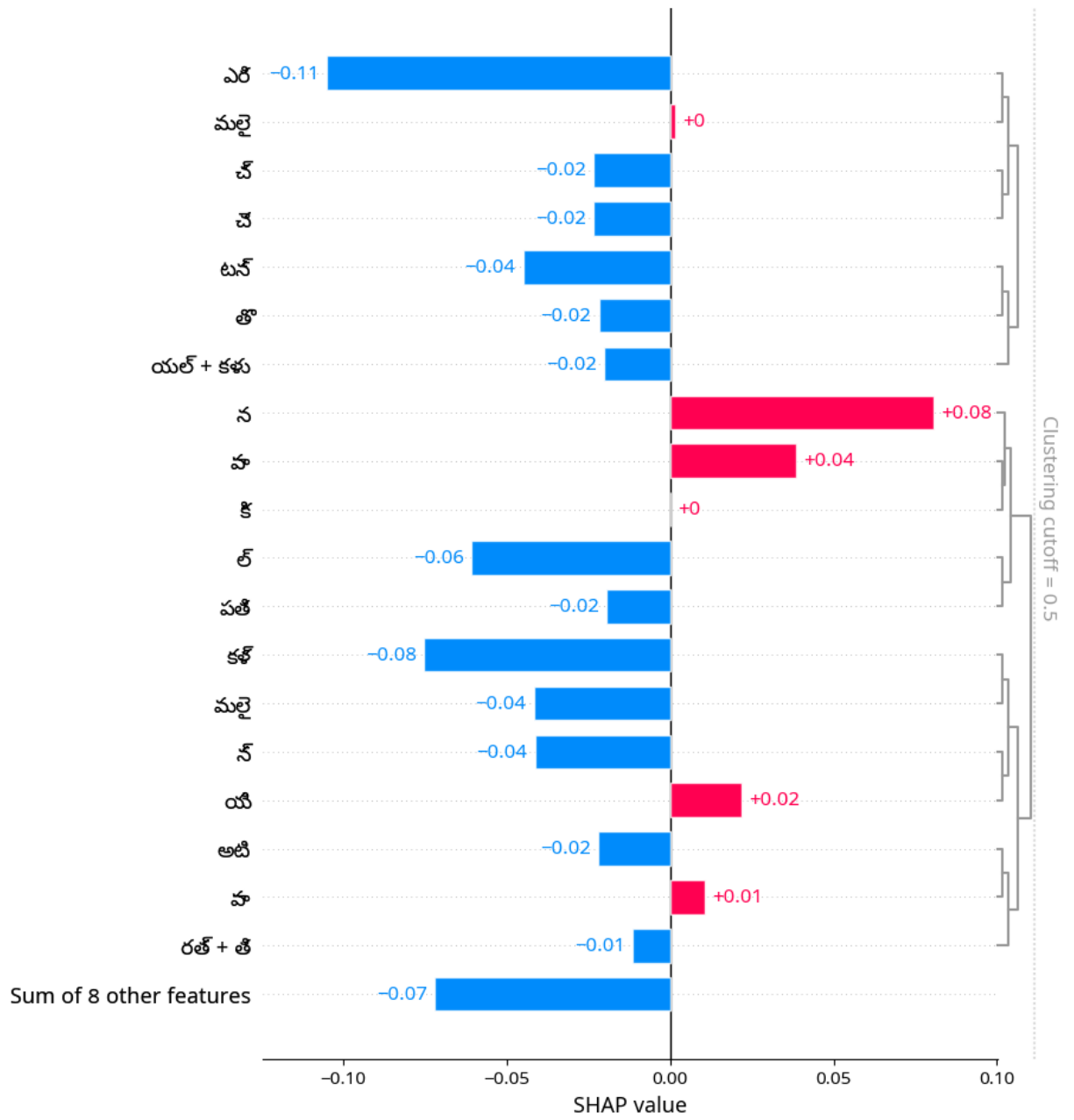


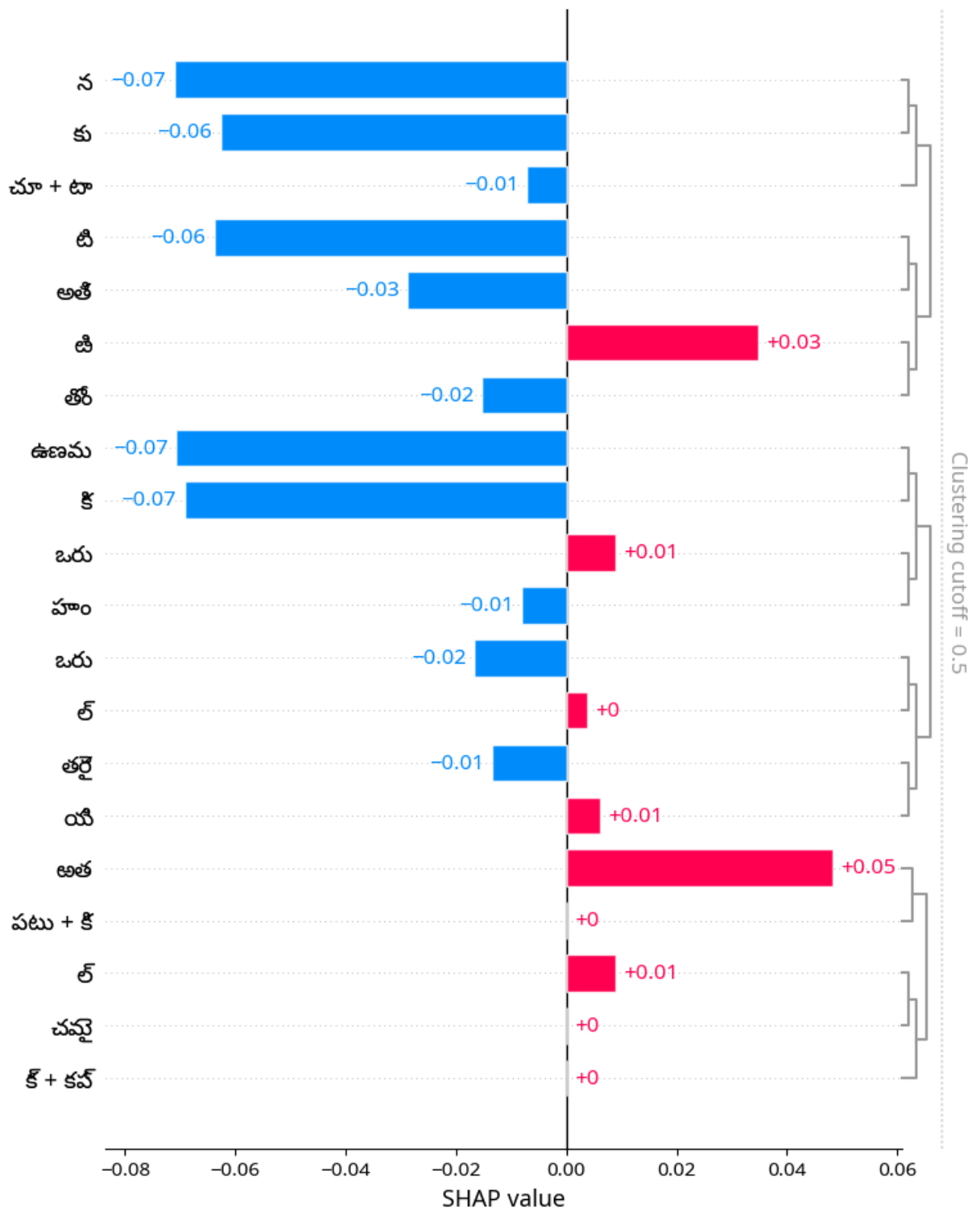


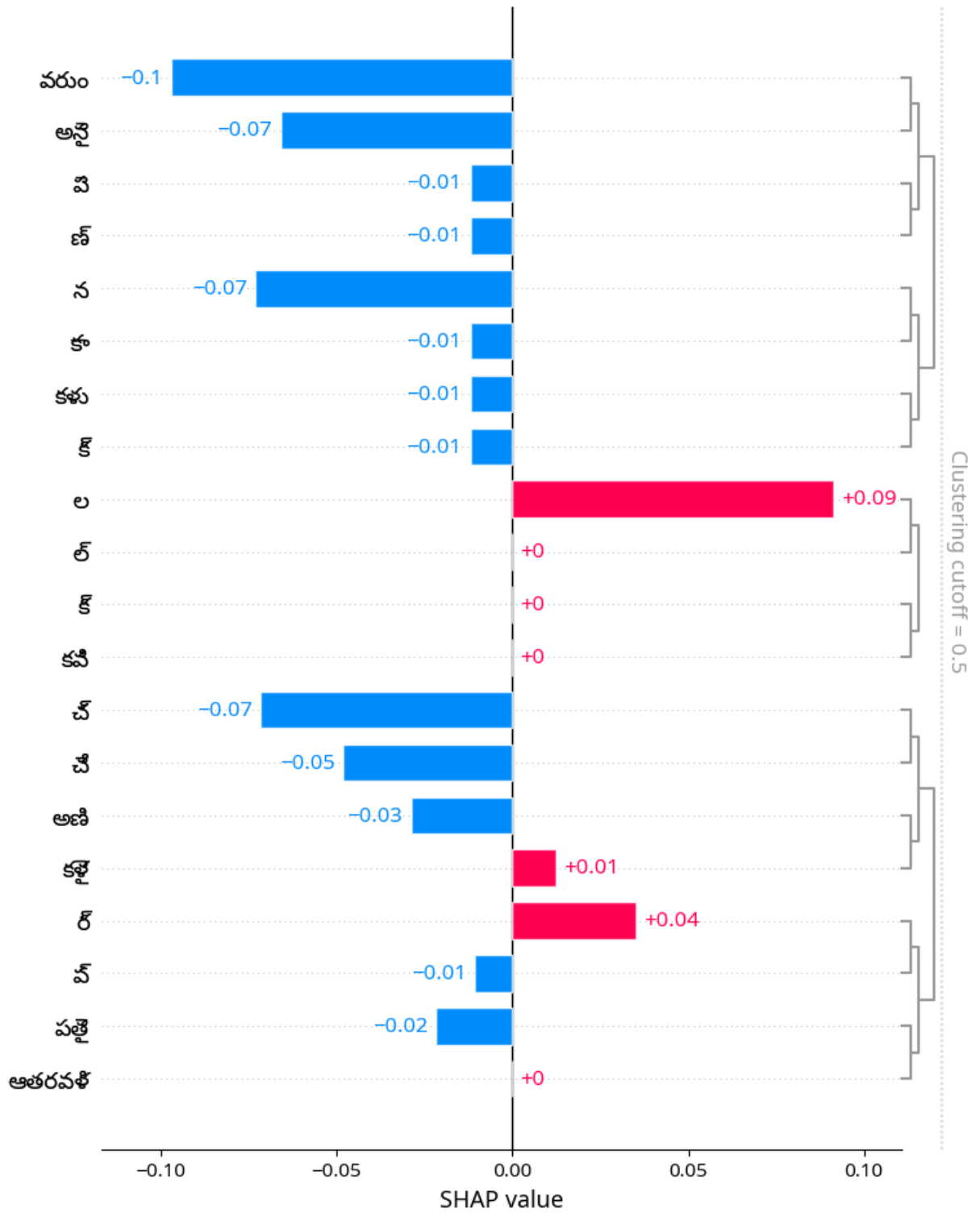












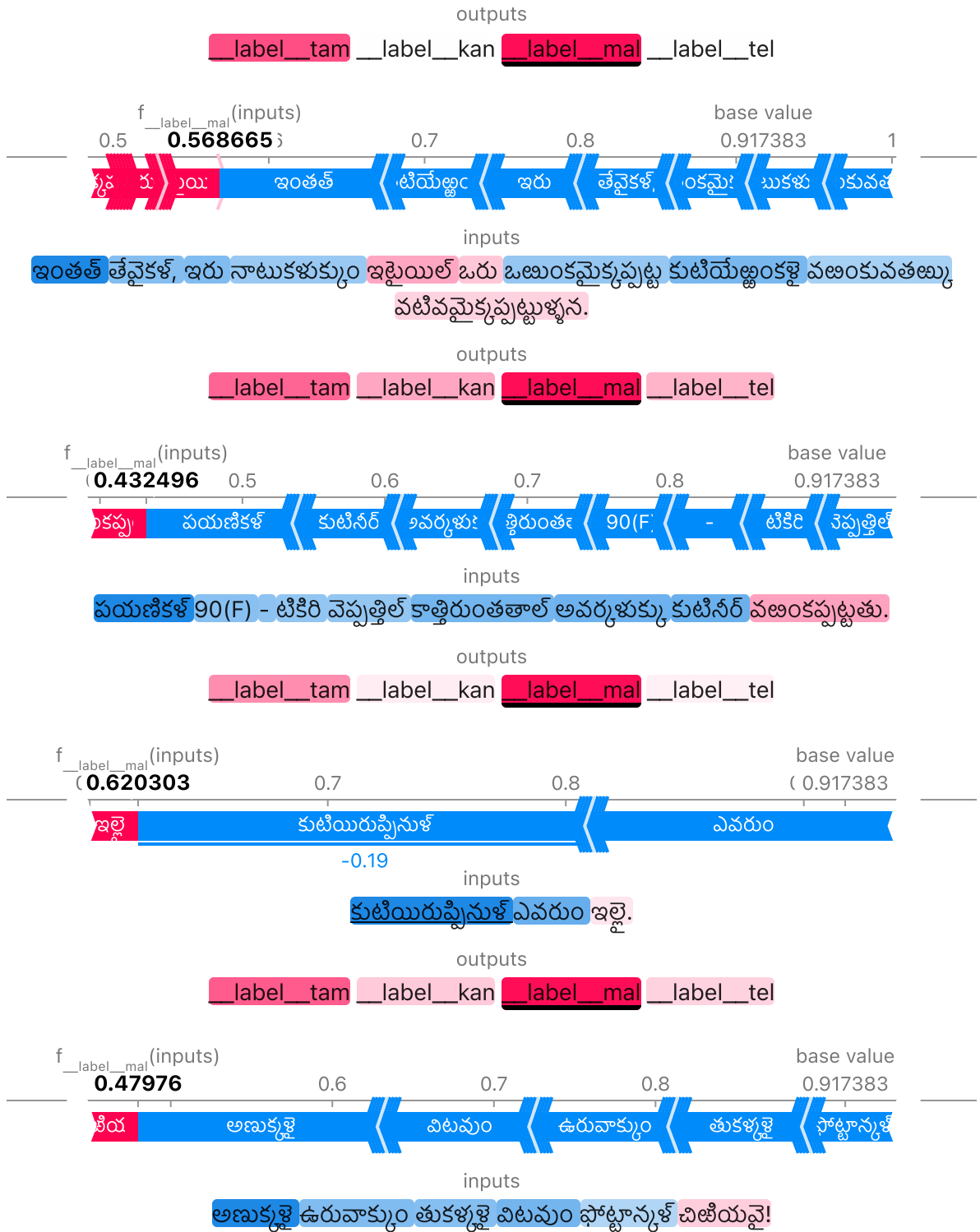
Word Level Explanations

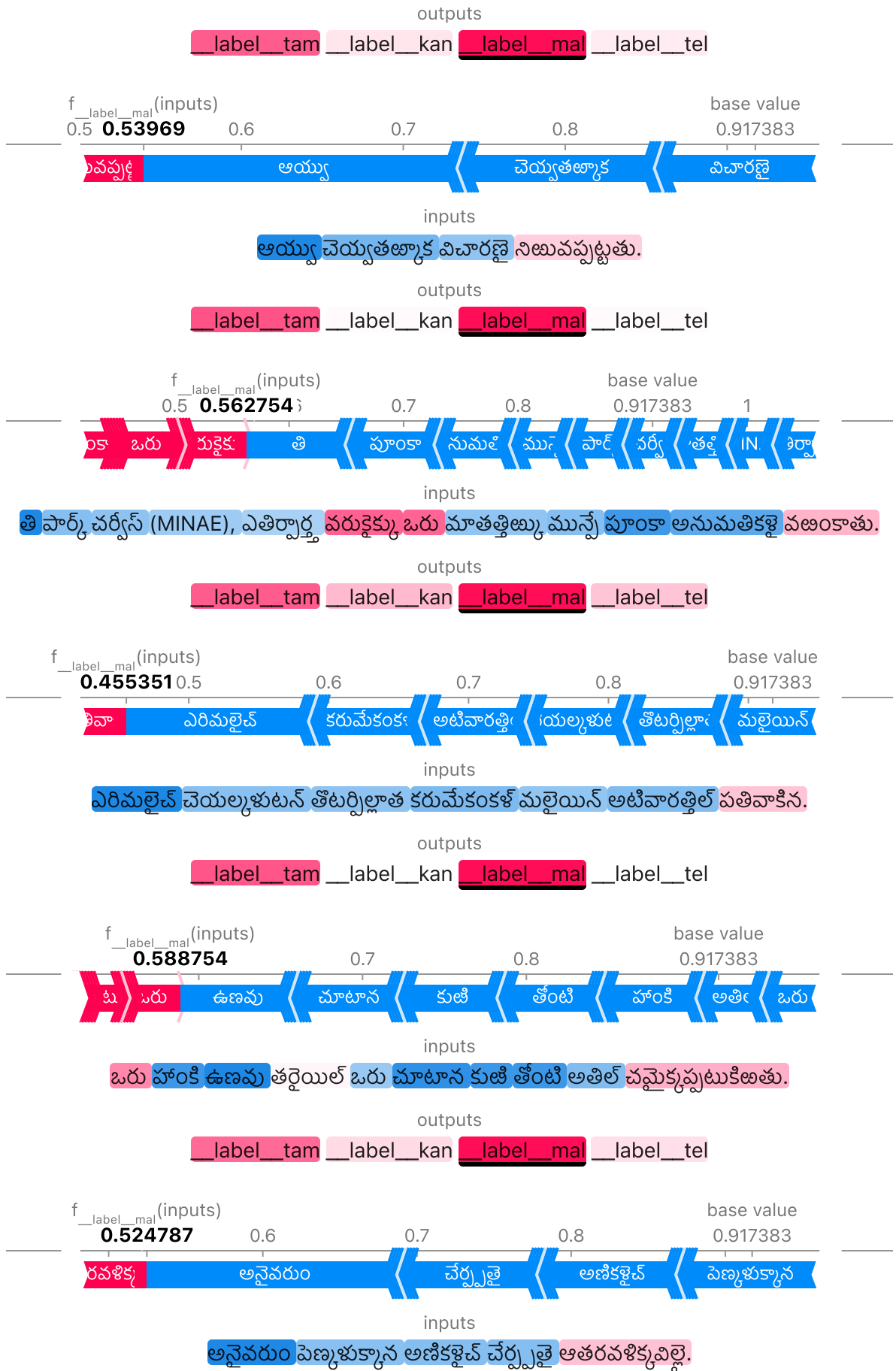
```
In [ ]: masker = shap.maskers.Text(r' ') # this will create a basic whitespace token
shap_explainer = shap.Explainer(predict_probability, masker, output_names=["shap_values"])
shap_values = shap_explainer(sentences)
```

PartitionExplainer explainer: 3989it [05:03, 12.74it/s]

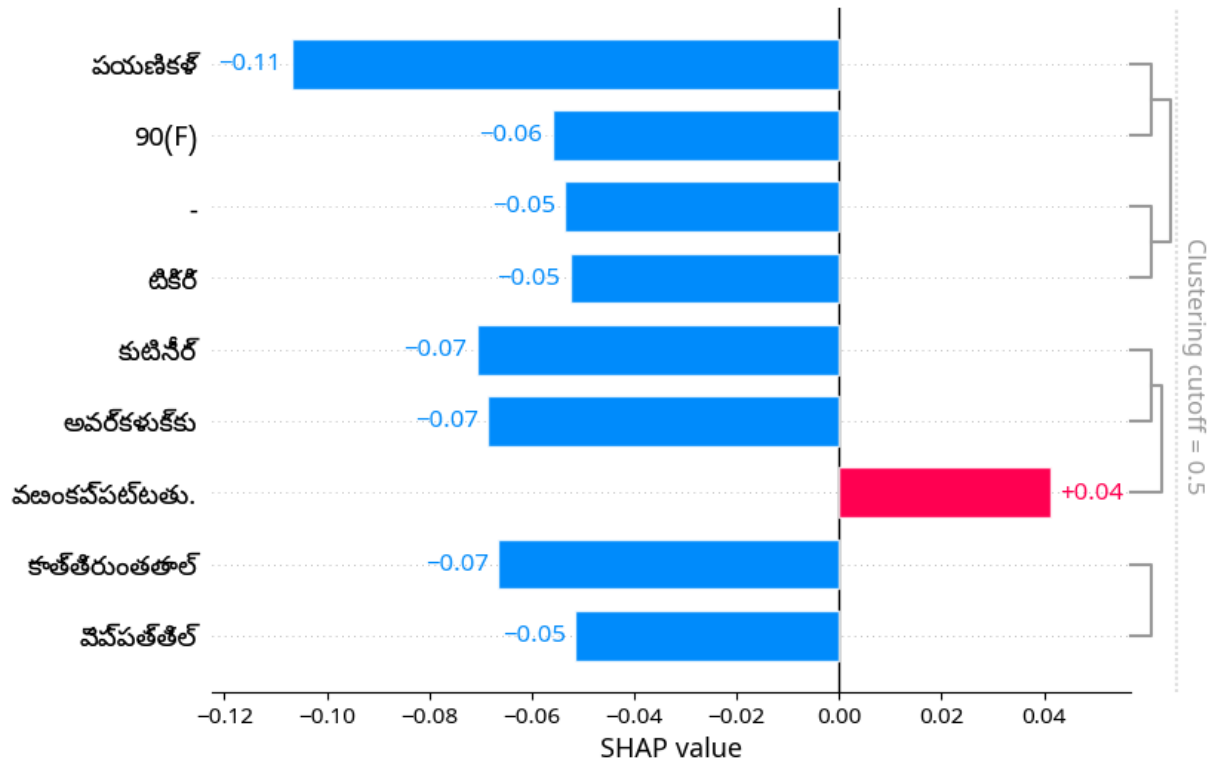
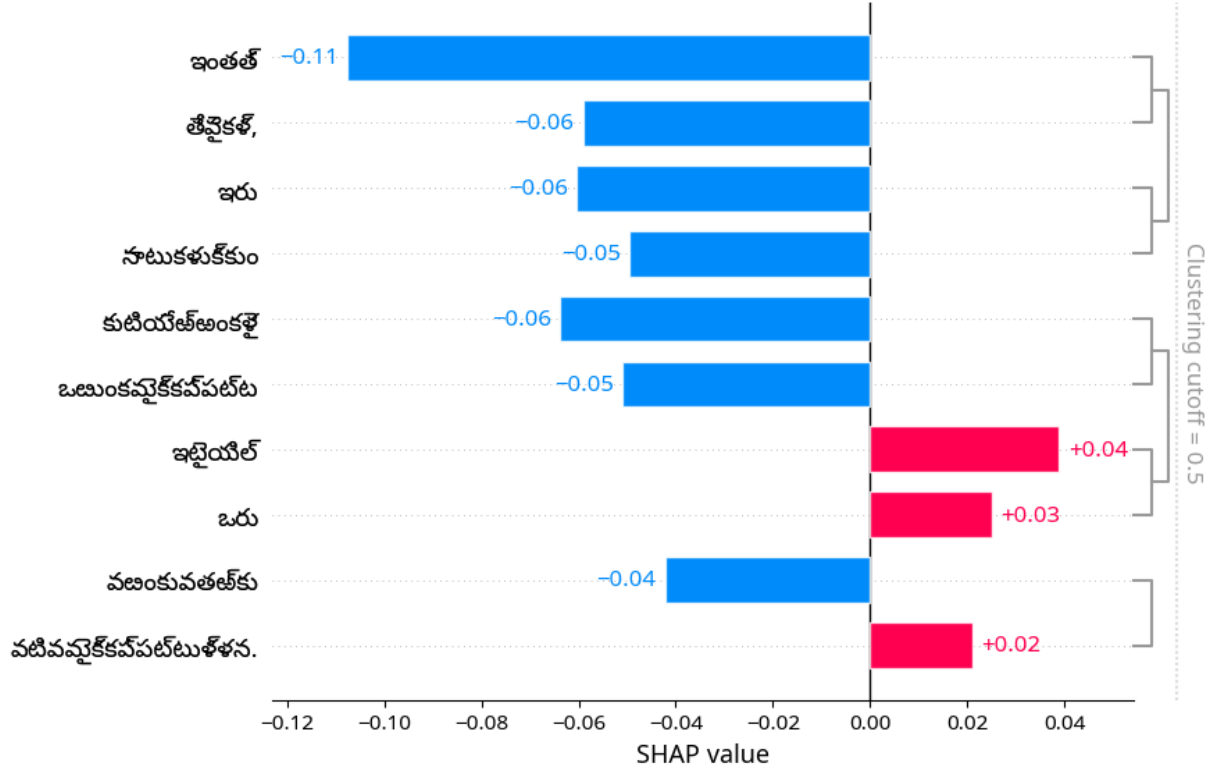
Explanations > 0.15

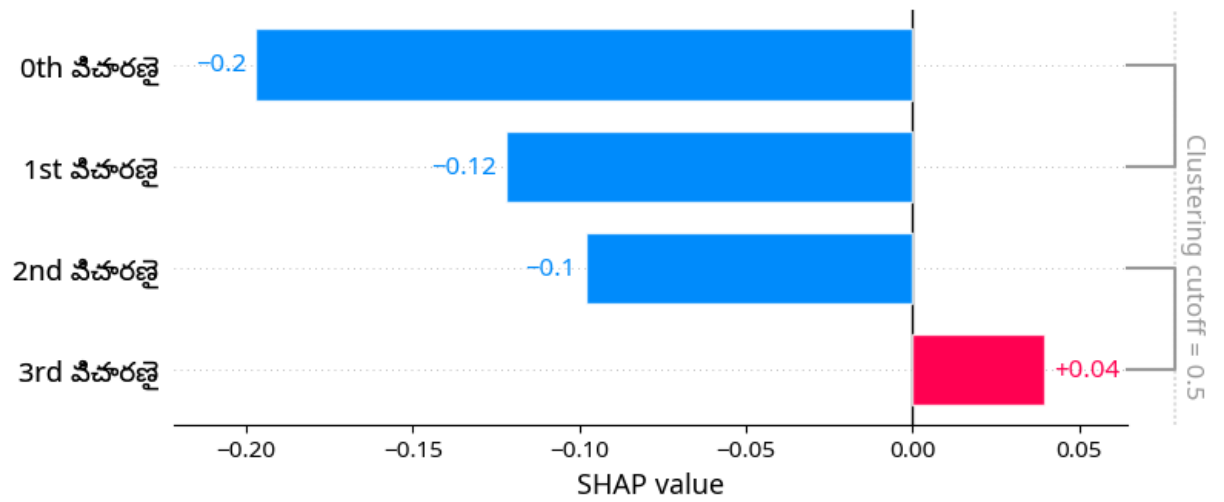
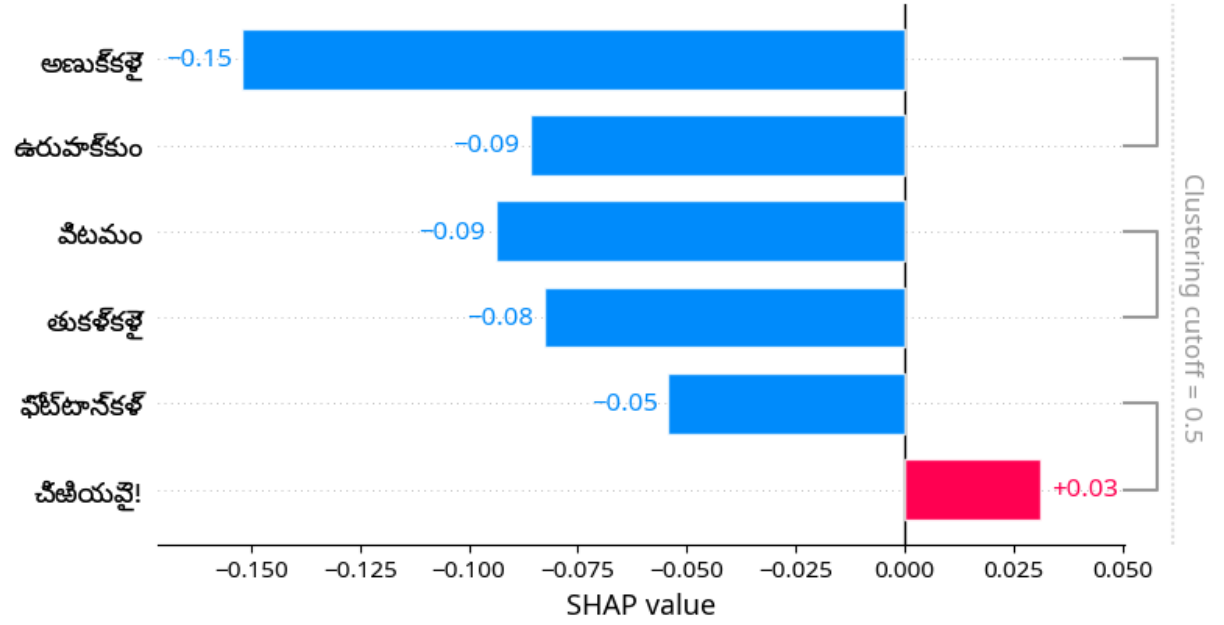
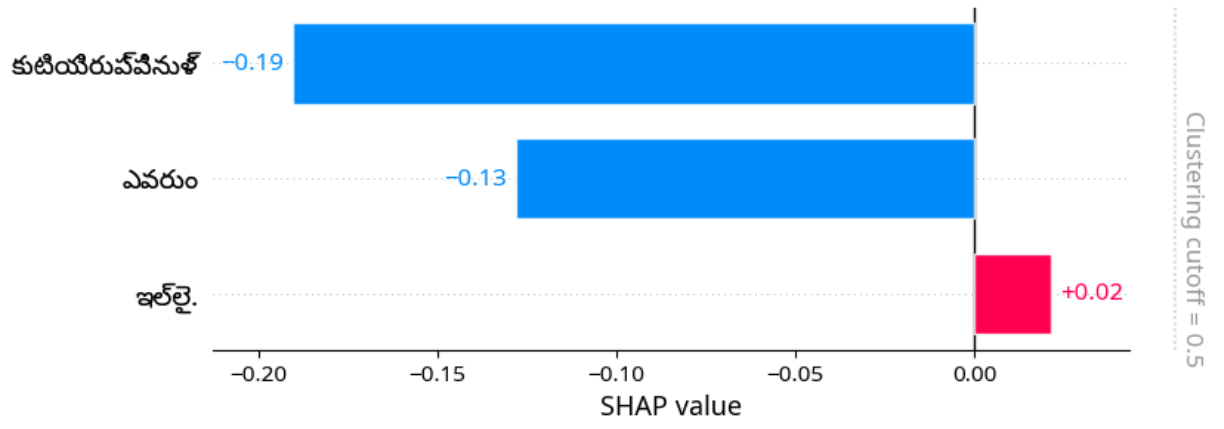
```
In [ ]: for i in ix_array:
        shap.plots.text(shap_values[i])
```

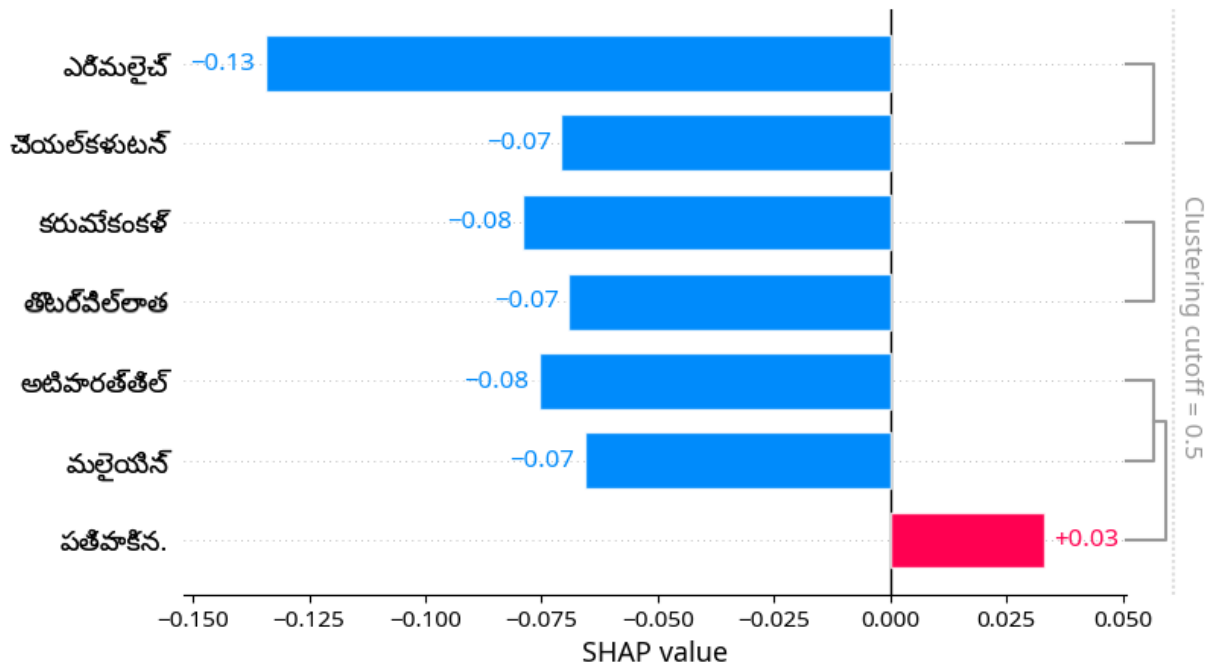
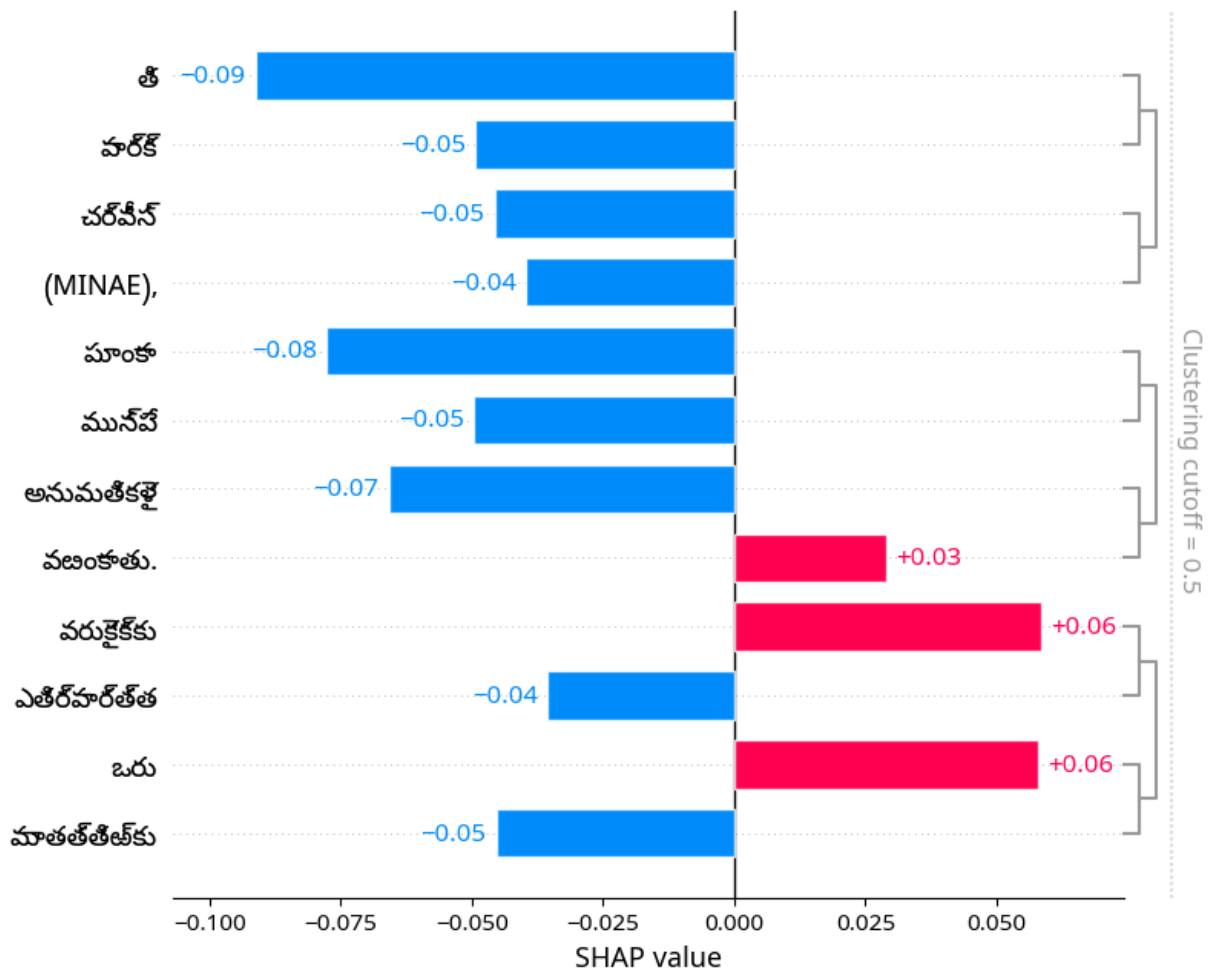


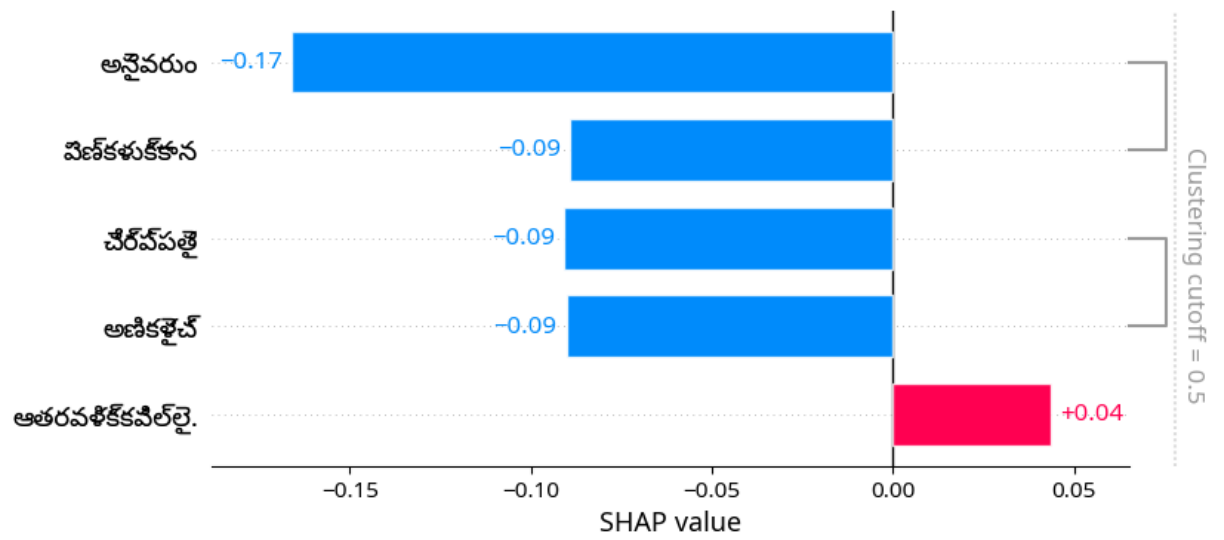
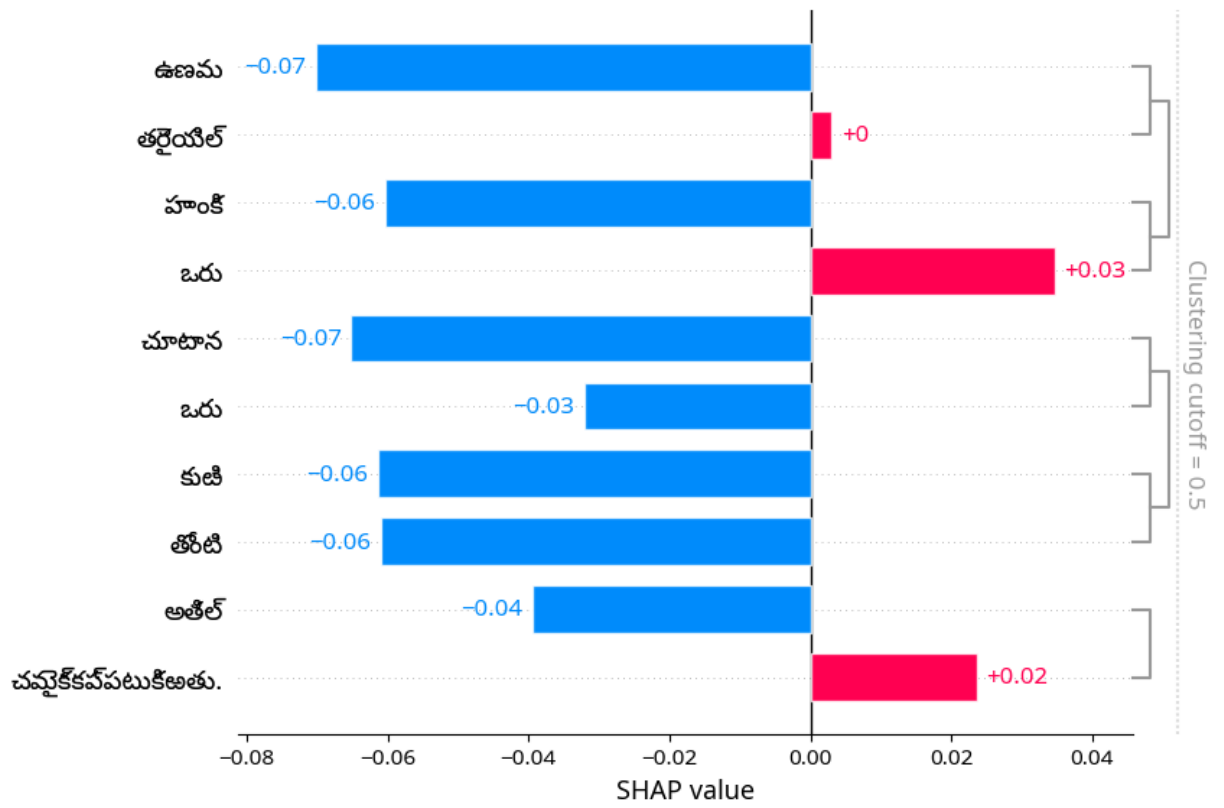


```
In [ ]: for i in ix_array:
        shap.plots.bar(shap_values[i][:,2], max_display=20)
```









```
In [ ]: for i in ix_array:
        shap.plots.bar(shap_values[i][:,0], max_display=20)
```

