```
from transformers import AutoTokenizer, AutoModelForSequenceClassification
from scipy.special import softmax


import pandas as pd
df2 = pd.read_csv('McDonald_s_Reviews.csv', encoding="ISO-8859-1")
df2.head()
```

	reviewer_id	store_name	category	store_address	latitude	longitude	rating_c
0	1	McDonald's	Fast food restaurant	13749 US-183 Hwy, Austin, TX 78750, United States	30.460718	-97.792874	
1	2	McDonald's	Fast food restaurant	13749 US-183 Hwy, Austin, TX 78750, United States	30.460718	-97.792874	
2	3	McDonald's	Fast food restaurant	13749 US-183 Hwy, Austin, TX 78750, United States	30.460718	-97.792874	

```
def clean(review):
    review = review.lower()
    review = review.replace('ï', '')
    review = review.replace('¿', '')
    review = review.replace('½', '')
    review = review.replace('ý', '')
    review = " ".join(review.split())
    review = review.rstrip()
    return review

df2['clean_review'] = [clean(review) for review in df2['review']]


#reviews = df2['clean_review']
import re

def remove_punctuation(text):
    return re.sub(r'[^\w\s]', '', text)
df2['clean_review'] = df2['clean_review'].apply(remove_punctuation)



reviews = df2['clean_review']


reviews[0]
```

```
'why does it look like someone spit on my food i had a normal transaction everyo
ne was chill and polite but now i dont want to eat this im trying not to think a
bout what this milky whiteclear substance is all over my food i d sure am not co
```

```
roberta = "cardiffnlp/twitter-roberta-base-sentiment"
model = AutoModelForSequenceClassification.from_pretrained(roberta)
model.to("cuda")
tokenizer = AutoTokenizer.from_pretrained(roberta,model_max_length = 514)

labels = ['Negative', 'Neutral', 'Positive']
```

```
from tqdm import tqdm
res= []
i = 0
for review in tqdm(reviews):
  try:
    encodedReview = tokenizer(review, return_tensors='pt').to('cuda')
    output = model(**encodedReview)
    scores = output[0][0].detach().cpu().numpy()
    scores = softmax(scores)
    res.append(scores)
  except Exception as e:
    print(e)
    i +=1
    print(i)
    res.append(None)

  # for i in range(len(scores)):
  #   l = labels[i]
  #   s = scores[i]
  #   print(l,round((s*100),2))
```

```
 52%|█████     | 17325/33396 [05:12<04:36, 58.02it/s]The expanded size of the tensor (566) must match the existing size (514
1
```

```
res[0]
```

```
array([0.9752794 , 0.02260427, 0.00211631], dtype=float32)
```

```
 71%|███████   | 23700/33396 [07:03<02:37, 61.56it/s]The expanded size of the tensor (568) must match the existing size (514
```

```
import pandas as pd
ress = pd.Series(res)
```

```
 84%|████████  | 28103/33396 [08:21<01:30, 58.40it/s]The expanded size of the tensor (523) must match the existing size (514
```

```
ress.to_csv('senimentsBertAllThree.csv')
```

```
7
```

```
ress
```

```
0         [0.9752794, 0.022604272, 0.002116306]
1        [0.002685675, 0.023319323, 0.97399503]
2          [0.7752055, 0.21287917, 0.011915322]
3         [0.005768352, 0.09304767, 0.90118396]
4          [0.92126226, 0.07193809, 0.00679967]
                        ...
33391    [0.9745224, 0.023127828, 0.0023498212]
33392     [0.0021946663, 0.01530327, 0.9825021]
33393       [0.2315547, 0.69735193, 0.07109343]
33394     [0.58317536, 0.33397937, 0.082845315]
33395      [0.005174789, 0.10072993, 0.8940953]
Length: 33396, dtype: object
```