## 1. Description of dataset used

The dataset used for this task is customer reviews for Amazon products bought. Data file is csv file with 24 columns. Although we are only using column 'reviews_text' for sentiment analysis task and then I only used one column from the actual dataset for further analysis.

```
RangeIndex: 5000 entries, 0 to 4999
Data columns (total 24 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   id                    5000 non-null   object 
 1   dateAdded             5000 non-null   object 
 2   dateUpdated           5000 non-null   object 
 3   name                  5000 non-null   object 
 4   asins                 5000 non-null   object 
 5   brand                 5000 non-null   object 
 6   categories            5000 non-null   object 
 7   primaryCategories     5000 non-null   object 
 8   imageURLs             5000 non-null   object 
 9   keys                  5000 non-null   object 
 10  manufacturer          5000 non-null   object 
 11  manufacturerNumber    5000 non-null   object 
 12  reviews.date          5000 non-null   object 
 13  reviews.dateAdded     1052 non-null   object 
 14  reviews.dateSeen      5000 non-null   object 
 15  reviews.doRecommend   5000 non-null   bool   
 16  reviews.id            29 non-null     float64
 17  reviews.numHelpful    5000 non-null   int64  
 18  reviews.rating        5000 non-null   int64  
 19  reviews.sourceURLs    5000 non-null   object 
 20  reviews.text          5000 non-null   object 
 21  reviews.title         4987 non-null   object 
 22  reviews.username      4999 non-null   object 
 23  sourceURLs            5000 non-null   object 
dtypes: bool(1), float64(1), int64(2), object(20)
```

## 2. Details of preprocessing text.

The text is preprocessed (cleaned and transformed) using NLP techniques to ensure it's in a suitable format for analysis. This might involve spaCy for advanced NLP tasks.

As I only copied 'reviews.text' column from original dataset.

I. Firstly dropped all rows with missing values( no records )

```
reviews_data = df[['reviews.text']].dropna()
```

2. Then removed all punctuations using `.is_punct()`

3. Filtered all stopwords from text using `.is_stop()`

4. Then converted the text into lower case using function .lower()

5. Lemmatised and tokenised all rows for further analysis

```
return ' '.join([token.lemma_.lower() for token in doc if not token.is_stop and not token.is_punct])
```

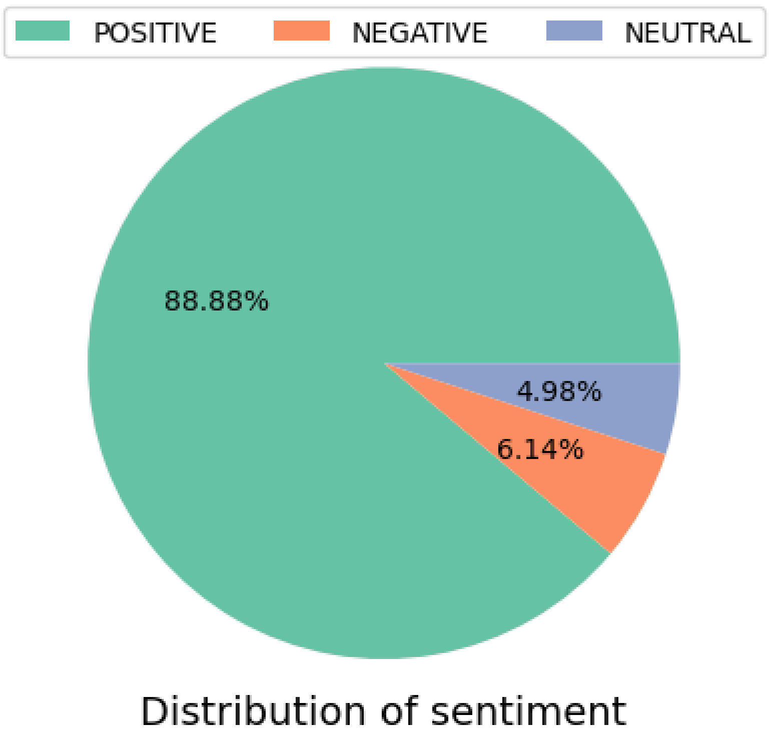
## 3. Evaluation of results

The processed data then used for evaluation based on polarity and sentiment of text

1. The preprocessed text is converted into a numerical format that the model understands.
2. **Sentiment Scoring:** The ML model uses the extracted features to calculate a sentiment score based on the patterns it learned during training.
3. The sentiment score is then mapped to a polarity (positive, negative, neutral) based on predefined thresholds.

```
# Function to analyse polarity score of reviews

def analyze_polarity(text):

# Analyze sentiment with TextBlob

    blob = TextBlob(text)

    polarity = blob.sentiment.polarity

    return polarity


# Function to get sentiment

def polarity_score(polarity):

    if polarity > 0:

        sentiment = 'positive'

    elif polarity < 0:

        sentiment = 'negative'

    else:

        sentiment = 'neutral'

    return sentiment
```

```
def polarity_strength(polarity):
    if polarity == 1:
        strength = 'Very Positive'
    elif (polarity > 0 and polarity < 1):
        strength = 'Positive'
    elif polarity == 0:
        strength = 'Neutral'
    elif ( polarity < 0 and polarity > -1 ):
        strength = 'Negative'
    elif polarity == -1:
        strength = 'Very Negative'
    return strength
```



## 4. Insights into model’s strengths and limitations

In the vast digital marketplace of Amazon, customer reviews serve as a treasure trove of insights. Leveraging Natural Language Processing (NLP) in data science allows us to unravel the sentiments, opinions, and trends hidden within these reviews.

Sentiment Analysis Magic:

- NLP excels at sentiment analysis, providing a nuanced understanding of customer emotions. Analysing Amazon product reviews through sentiment analysis unveils the emotional tone associated with each product, helping businesses gauge overall customer satisfaction. This invaluable insight aids in product improvement and tailoring future releases to align with consumer preferences.

Customer Experience Enhancement:

- Through NLP, businesses can identify common issues raised by customers, enabling proactive measures to enhance customer experience. This can include addressing recurring concerns, improving product descriptions, or even optimising the user interface.

Limitations:

Contextual Ambiguity:

- NLP struggles with contextual ambiguity and sarcasm, making it challenging to accurately interpret the intended meaning in some reviews. This limitation may lead to misinterpretations, affecting the reliability of the analysis.

Bias and Diversity Challenges:

- Amazon's customer base is diverse, and so are their linguistic expressions. NLP models may not be universally accurate across all demographics, potentially leading to biassed interpretations. For instance, regional colloquialisms or cultural nuances may be misunderstood.

Ever-evolving Language:

- Language is dynamic, with new slang and expressions emerging constantly. NLP models may struggle to keep up with these linguistic shifts, potentially leading to misinterpretations or outdated analyses.

Conclusion:

Navigating the intricate landscape of Amazon product reviews using Natural Language Processing offers a treasure trove of insights. However, it's essential to acknowledge the method's strengths and limitations. Striking a balance between the capabilities of NLP and the complexities of human language is key to harnessing its full potential for data-driven decision-making in the dynamic world of e-commerce.