# Task 26 - Sentiment analysis with spaCy

In this task, we developed a Python program that uses the spaCy and TextBlob libraries to perform sentiment analysis on a dataset of Amazon product reviews.

## A description of the dataset used

The dataset is a list of over 34,000 consumer reviews for Amazon products like the Kindle and Fire TV Stick. The dataset includes: basic product information, date of review, rating, review text, and more for each product. For this task we will focus on the review text variable.

## Details of the preprocessing steps

The dataset was preprocessed by selecting the review text column and dropping missing values. The text was then converted to lowercase and white space was also stripped from the beginning and end of each review.
Next, various ‘stop words’ were removed and the resulting text was lemmarised to convert each word to its root form. Finally, any non-alphanumeric characters were removed from the reviews.

## Evaluation of results

The sentiment of the reviews were analysed using the TextBlob natural language processing library. We took a random sample of 5% of the data and calculated the sentiment polarity for each review, this was then plotted against the original review rating, see Figure 1.

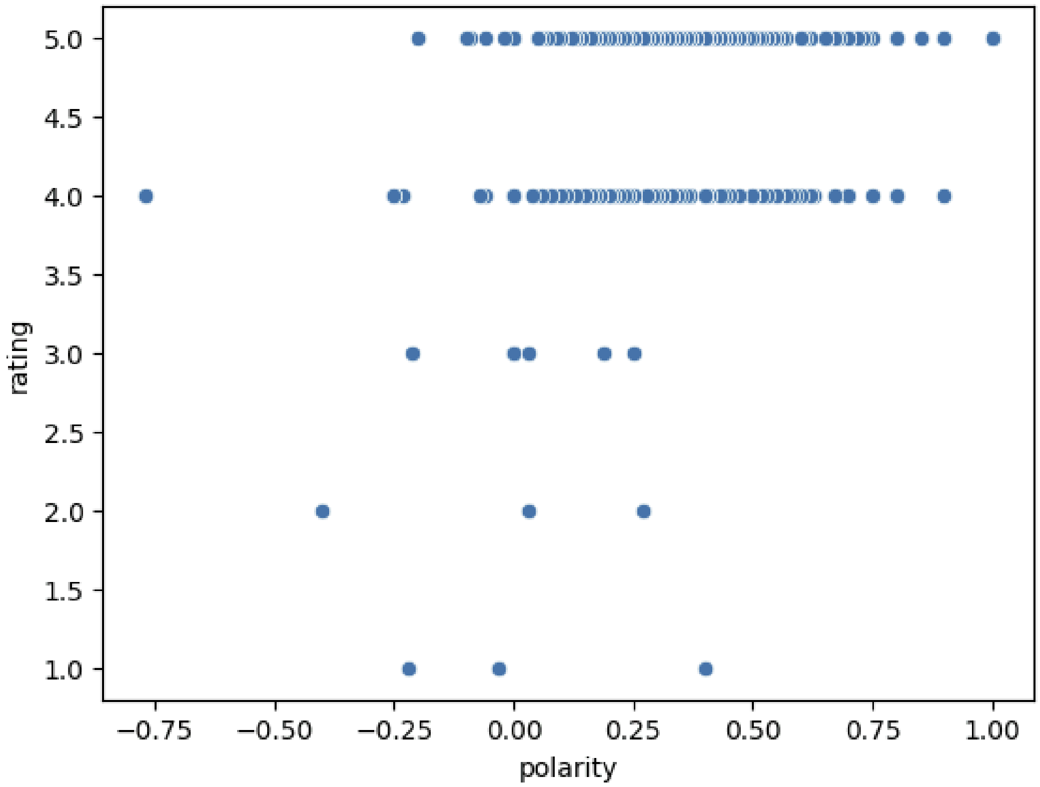

Figure 1. Shows a scatterplot of the TextBlob sentiment polarity against the user given rating. A polarity of 1 relates to a highly positive sentiment and a polarity of -1 relates to a highly negative sentiment.

The scatterplot shows that if the sentiment polarity is positively correlated with positive reviews, in particular, if the polarity is greater than 0.5 then the user given rating is either 4 or 5.

## Insights into the model's strengths and limitations

The model is good at determining whether a review is positive, that is, high polarity implies a high rating. However, a low polarity does not necessarily lead to a low user defined rating. The most likely cause of this is the misreading of negation as the word ‘not’ is included as a stopword and so the overall sentiment gets misread.

For example in the following review the first sentence is preprocessed to ‘*price tablet bad’.*

*Review index 1298 - sentiment polarity -0.77 - user rating 4.*
*for the price, this tablet is not bad. i found a couple of things that is a bit annoying. every time you turn on the device, ads will appear in the lock screen. also the picture gallery, it takes several minutes to load. the wait is very annoying.*

Further analysis on the correct stopwords and preprocessing steps will be useful to enhance the model.