1. Dataset Description

The dataset used in this project is the **Datafiniti Amazon Consumer Reviews of Amazon Products (May 2019)**, sourced from Kaggle. It contains detailed customer reviews of various Amazon products, including fields like product names, brands, prices, and — most importantly — **text-based customer reviews** (reviews.text). This column serves as the input for our sentiment analysis task.

2. Preprocessing Steps

Preprocessing is a critical step in any natural language processing (NLP) task. Here's what was done:

Missing value removal: I used dropna() to ensure only complete reviews were analysed.

Lowercasing: All text was converted to lowercase to ensure uniformity (e.g., “Great” and “great” are treated the same).

Stop word removal: Common filler words (like “the”, “is”, “on”) were removed using spaCy’s is_stop attribute, as they add little to sentiment meaning.

Token filtering: Only alphabetic tokens were retained (is_alpha) to eliminate punctuation and numeric noise.

This cleaned version of each review was then passed to the sentiment analysis function.

3. Evaluation of Results

I tested the model on a random sample of product reviews. The results generally aligned with the tone of the reviews:

Highly positive reviews (e.g., “I love this product”) were rated as Positive.

Complaints or critical feedback (e.g., “terrible service”) were correctly flagged as Negative.

Mixed or factual statements tended to yield Neutral scores.

4. Insights: Strengths & Limitations

Strengths:

Lightweight and fast — perfect for exploratory sentiment analysis.

Easy to integrate with any pandas DataFrame.

Handles general language well.

Limitations:

Not domain-specific — may misinterpret jargon, sarcasm, or mixed-tone reviews.

Cannot detect intent or context beyond surface sentiment.

Could be improved with domain-specific fine-tuning or deep learning models.