Intel Unnati Project Report

Project Title: Intel Products Sentiment Analysis from Online Reviews

Web Scraping with Selenium

Web scraping is when one automates the collection of content and data from websites by extracting the underlying HTML code. This technique differs from screen scraping, which captures what is displayed visually on the screen. The importance of data analysis has grown alongside the availability of huge volumes of raw data hence resulting in development of specialized Python packages that simplify web scraping.

Selenium as an open-source project provides a range of tools and libraries for browser automation. Among them is a playback tool (Selenium IDE) that allows creation of test for most modern browsers without any knowledge about specific scripting languages necessary for testing. With Selenium web scraping, automated gathering of essential information through Selenium WebDriver Browser Automation is possible.

For example, Selenium web scrapers can extract data from following Amazon review pages for Intel products:

Intel Core i5-12500 Processor

Intel i3-13300 Desktop Processor

Intel i5-13500 Desktop Processor

Intel i7-13700 Desktop Processor

Using selenium for web scraping makes the extraction process easier and speeds up collecting and analysing review data from several pages.

Data Preprocessing

Features Offered

The dataset has columns named title, description and sentiment. Usually, in the title column, a brief description is given of the review provided by the client that may often indicate either product names or satisfaction degree. The description column is for writing more about the product; it includes things like features, performance, quality and customer experiences and

comments on another level. It goes further to provide an analysis based on usability, reliability among other factors as well as giving an overall evaluation of what customers think.

This project uses TextBlob to analyze sentiment from different Intel products reviews. Reviews are classified into positive, negative or neutral by polarity scores using TextBlob: a score above 0 indicates positive while below zero represets negative while a score equaling zero means neural. It also allows user to identify whether each review was positive, negative or neutral through its broad sentiment score.

Loading the Data(Data Collection)

Various Amazon links were used to obtain Intel product reviews. This data was scraped from multiple pages and loaded into a CSV file for more analysis purposes. The merged_amazon_reviews.csv file consists of columns – title; rating and description.

On this ground, rating column which shows overall rate given by customer has been removed due to repetition.

A sentiment column was added to the dataset using TextBlob so as to show if any review is positive or not (or if some are neutral). The words in the review help determine its sentiment. TextBlob detects and classifies the sentiment, enabling a count of positive and negative reviews to assess the product's overall sentiment.

Data Cleaning

The raw data extracted from Amazon review pages is cleaned for efficient processing. Unwanted elements such as lowercase letters and special characters are removed. Non-English reviews are eliminated, and English reviews are corrected for grammatical and vocabulary errors. Common English words like "a" and "the" are removed. Emojis are converted to text format.

Exploratory Data Analysis

This dataset is organized into title, description and sentiment columns where TextBlob uses it to classify Intel product reviews into positive, negative or neutral categories based on polarity scores.
Exploratory Data Analysis (EDA) also includes sentiment score distributions analysis, word frequencies and correlations between review length and sentiment. These visualizations include histograms, bar charts, and word clouds which show patterns and common themes in the data. EDA provides useful insights into pattern of the data for guiding further analysis and modeling.

Random Forest Classification

A powerful and flexible machine learning algorithm known as Random Forest Classification (RFC) is what this paper is about.

It involves creating multiple decision trees during training phase and obtaining the mode of the classes (classification) of the individual trees. This ensemble approach helps to reduce overfitting and increase accuracy hence making RFC one of the most effective algorithms in a wide variety of projects.

The RFC technique operates by constructing a set of randomly created decision trees that are each based on random subsets of data. During classification, each individual tree produces a class prediction, with the final prediction being determined by majority voting across all trees. This method improves this model's robustness together with its accuracy by mitigating any variance that may be caused by single tree.

In this project, we have chosen RFC because it is more efficient and performs better than other tools. Its ability to handle high-dimensional large datasets and its resistance against overfitting positions it as an appropriate choice. With RFC we have been able to achieve high accuracy and reliability in our classification tasks so that precise and dependable results can be obtained.

Conclusion

Our analysis on customer reviews of Intel products benefited from the use of Selenium for web scraping, TextBlob for sentiment analysis and RFC for classification. From this analysis, we have been able to obtain insights that give an overall picture of how customers feel about Intel's products. These insights are crucial in making informed business decisions that improve product strategies. Additionally, this project demonstrates the need to marry web scraping with advanced data analysis techniques in order to extract meaningful insights from big data.