**Amazon Product Headphone Analysis**

**Data 608: Project Proposal**

(**Sneha Arora, Yongpeng Fu, Stuart Finley, Lu Li**)

**Purpose:**

Amazon is a platform where we can find multiple brands for all types of products. It’s a place that connects the buyer and the seller, where anyone can sell their product. Each product has a detailed information along with the ratings and reviews given by its customers. Users rely on this information provided to make relevant choices while buying a product of their choice.

The major purpose of this project is to create a big data application for Amazon Headphone Products. The focus of our analysis is prediction of price based primarily on the product reviews and ratings. Amazon provides an overall rating of a product item based on customer rating. But the reviews written by each customer are there for manual analysis only. We are analysing amazon product – ‘headphones’, using many features such as sound quality, battery and reviews to find the most optimal headphone according to the need of a user.

The major purpose of this project is to create a big data application for Amazon Headphone Products. The focus of our analysis is to use our web scrapped data to analyze what customers are saying about Headphone devices, discover insights into customer reviews, and potentially predict price based on reviews, ratings, and many distinct features like Volume control, Noise cancellation, and Connectivity technology. Eventually, we will build a website interface to show our analyse to the public.

**Data**

*What is your dataset(s)? Where can you find it, what do you think the strengths and weaknesses of this dataset are?*

(please add a little about data, the size of data and other problems faced during web scraping such as denial to access the page.etc.)

We use *web scraping* tool – ‘beautifulsoup’ to extract *price, description, review date, customer reviews and ratings* for each product. This would require developing a web scraper that can navigate the UK Amazon website and collect the desired data.

Following are the columns we extract:

|  |  |
| --- | --- |
| Product Name | The product name |
| Review Title | The title of each review |
| ASIN Number | Unique number given to each product |
| Price | The price of each item of the product |
| Description | Small description of each item given (for eg. Wireless / in-ear headphones etc.) |
| Review date | The date when the customer wrote the review for a item (Each review has a different review date) |
| Customer reviews | Reviews written by the customer about how they like/dislike the product. |
| Customer Rating | Rating given by the customer out of 5 for a particular item, giving a quick idea of the overall performance of the product from their perspective. |

**Methodology**

*If machine learning - what is the story you will tell? How will you determine if the machine learning was successful?*

(add more to respective parts)

Web scraping: Develop a generalized web scraper to retrieve contents from Amazon website.

Database storage using AWS: Store data in cloud - explore relational vs nosql database

Sentiment Analysis: Data cleaning and wrangling to remove the stop words, blanks and other information not required in the review data scraped from Amazon website. Perform sentiment analysis on the reviews of the products to derive the review rating – required for price prediction. Also, to get the summarisation of reviews per item of the product.

Predict the price of a new item for a business user or a customer using the features above in addition to the review rating and summary.

Dash Website Interface: To show our analysis and expose our general-purposed web scrapper algorithm, we will use Dash to build a simple one-page web application. We will use dash bootstrap components, and Dah Mantine components to build the layout and add interactive components (joined effort between teams). The final application environment will be stored in a docker container to be deployed in the Amazon EC2.

**Individual learning objectives:**

*For example: "I will deploy my own Docker container onto a Google Cloud instance by March 20."*

Yongpeng Fu:

* I will have an understanding on HTML, CSS, Flask framework, and build the Dash web foundation by 15 March.
* Deploy our Plotly/Dash web using Docker and Amazon EC2 by 1 April.

Stuart Finley:

Lu Li:

Sneha Arora:

* Understand sentimental analysis and create review summarization using Abstractive summarization technique with spacy for each item of the product by 15 March.
* Generate review rating using sentiment analysis and compare with the customer rating. Find the difference to give a realistic picture told by the reviews by 1 April.

**Feedback Received (Pitch):**

Overall Feedback

Good work with just some minor points for improvement.

I really like that there are clear indications that you're already working creating your system and working to understand what will work and what won't.

Also remember to keep an eye on minor details - the technical term is "sentiment analysis" rather than "sentimental analysis."  I think the typefaces you used for your slides changed a little bit throughout your pitch - always helpful to do a quick check for these little things!

The project seems reasonably sized. Good work!

**References**

* Gangwar, M. (2022) *How to scrape Amazon product information using beautiful soup*, *DigitalOcean*. DigitalOcean. Available at: https://www.digitalocean.com/community/tutorials/scrape-amazon-product-information-beautiful-soup (Accessed: March 2023).
* *Fake-useragent* (no date) *PyPI*. Available at: https://pypi.org/project/fake-useragent/ (Accessed: March 2023).
* Jhnwr, J. (2020) *Scrape-amazon-reviews/review-scraper.py at main · JHNWR/Scrape-Amazon-reviews*, *GitHub*. Available at: https://github.com/jhnwr/scrape-amazon-reviews/blob/main/review-scraper.py (Accessed: March 2023).
* Bhatt, A., Patel, A., Chheda, H. and Gawande, K., 2015. Amazon review classification and sentiment analysis. *International Journal of Computer Science and Information Technologies*, *6*(6), pp.5107-5110.
* Sblendorio, D. (2022) *How to do text summarization with Deep Learning and python*, *ActiveState*. Available at: https://www.activestate.com/blog/how-to-do-text-summarization-with-python/ (Accessed: March 2023).