CSCI 5302 - Final Project

Patrick Connelly   Aneesh Khole   Uttara Ketkar

2024-03-10

Table of contents

1 Introduction

1.1 Concept and Motivation

Customers, when searching for products with specific features and aspects, need sufficient information to make a decision as to whether to procure a specific product. According to research by Guha Majumder, Dutta Gupta, and Paul (2022), if a customer can gather and understand product quality before the purchase, it is considered a search good, while experience goods are those which must be purchased or experienced to evaluate them. When a product is more in the directon of experience vs. search-based, other customers' experiences can shed light on its features and return on investment than information directly from the vendor can. Having reviews from reliable sources with sufficiently detailed information can enable greater confidence in a purchase, improved customer satisfaction, and smooth the process of ecommerce for customers.

We seek to expound upon the research of (Guha Majumder, Dutta Gupta, and Paul 2022) to explore additional recommended research areas to improve upon and increase the general applicability of the model.

1.2 What / Where

Guha Majumder, Dutta Gupta, and Paul (2022) provided the following summary model for what aspects and features they took into consideration in predicting the perceived usefulness of a customer review in Figure 1.1.

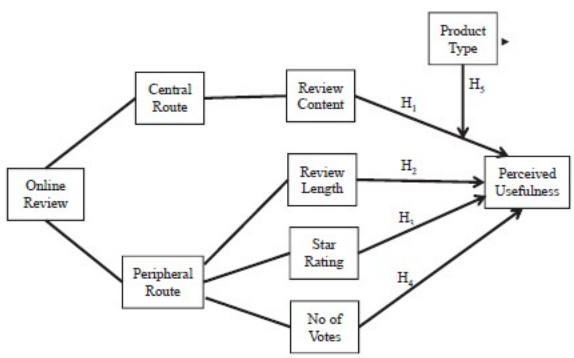

Figure 1.1: Model Overview

Furthermore, the authors provided the following areas for recommended additional research at the conclusion of their paper:

1. Expand the number of products beyond 3 items (one search, one experience, one mixed) to better generalize the model.
2. Explore customer or reviewer metadata for classifying reviewer types to enhance model performance.

We seek to examine the above two above items, and to explore the possibility of assessing a scale for products to determine the extent to which they are a search or experience-based product. We further seek to inspect additional potential modifiers to the underlying model for statistical and operational applicability; we've sought out work from other research teams to identify potential methods we can leverage to pursue these ends.

- Determining the polarity of a customer review by employing a classifier such as Naive Bayes.

- Using Kansei engineering approaches to convert unstructured product-related texts into feature–affective opinions.
- Attempting to assess the reliability of a customer's review based on star-rating and a 'sentiment score' of their textual feedback.

Exploring methods employed within each of combinations of these research efforts, we will pursue potential improvements on the models outlined in Guha Majumder, Dutta Gupta, and Paul (2022). We will examine additional products and product types between multiple e-commerce websites (BestBuy, Target, Amazon). A summary of our explorations are depicted in Figure 1.2.

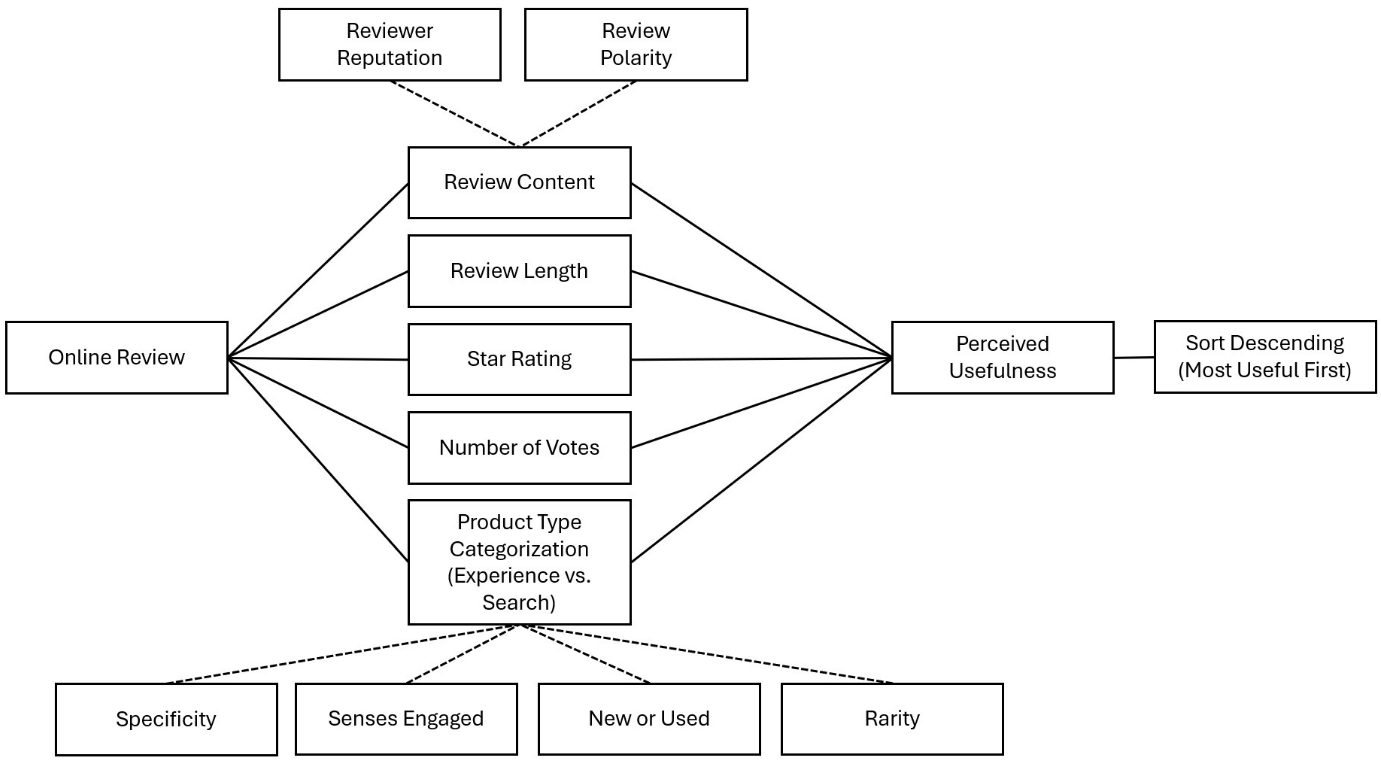

Figure 1.2: Model Modification Goals

This is not final, but what we plan to explore. If any metrics or measurements are found to not be significant in analysis and prediction of usefulness of a review, we will seek to explain the relationships (or lack thereof) and modify the final model accordingly. By incorporating these additional measures, we may be able to improve upon and generalize the original model to multiple product types across multiple e-commerce vendors.

1.3 Why It Matters

Feedback from customers can be beneficial to both vendors and consumers, but it is not always ordered by the most informative or beneficial feedback first. Certain features of products, reviews, and reviewers (such as reviewer reliability, review quality, product quality, specificity and detail of the product, amongst others) can impact the usefulness of the feedback on a customer-by-customer basis. Level of detail, star-rating, and number of votes that support the review as being useful to a customer can all help determine its usefulness to other customers. Leveraging metrics and data associated with a product, a review, and a reviewer together may allow for online vendors to improve consumer e-commerce experience, support identificaion of issues with product quality and sales, and enable vendors to adjust practices in product marketing, inventory, and manufacture.

Examining additional product types could support a generalization of the authors' methodology to other products. Furthermore, the exploration of a sliding scale for search vs. experience-based products can further support generalization and business goals. Producing a reliable scale and methods for classifying a products' degree of being experienced-based can inform vendors on:

- How to best sort product reviews
- Examine what are the most helpful reviews to know the performance of the product alongside customer experience and sentiment
- Adjust the product, its marketing, or future production based upon market efficacy
- Understand the emotions a customer wants to express through a review is crucial as it will affect the "recommendation score" of that particular product or a different one from a similar category.
  - To contribute in determining this recommendation score, we can use a probabilistic machine learning algorithm like Naive Bayes to determine the polarity (positive, negative, or neutral) of customer reviews.
  - Typically used for amending product design, Kansei Engineering can be used to incorporate human emotional responses into evaluation of a customer review.
- Determine which customer is trustworthy, meaning who has actually purchased the product versus a customer who gave a false review. Based on the 'customer reputation score', our aim is to classify customers into groups to judge reviewer reliability. This has two main aspects:
  - Star-rating score which is a discrete scale that tells the inclination of a customer.
  - Text review 'sentiment score' using NLP that explains customer opinions based on words.

1.4 Literature Survey

- Guha Majumder, Dutta Gupta, and Paul (2022)
  - Examined multiple-linear regression modeling to calculate the usefulness of an online review based upon type of product (search vs. experience), review sentiment, review star rating, review length, and number of votes for the review as being "useful". Suggested exploration using larger number of products as well as customer/reviewer metadata.
- Hu, Gong, and Guo (2010)
  - The proposed system employs a two-step process for opinion mining: identifying opinion sentences using a SentiWordNet-based algorithm and extracting product features from all reviews in the database. This feature extraction function focuses on identifying commonly expressed positive or negative opinions before extracting explicit and implicit product features.
- Rajeev and Rekha (2015)
  - This paper presents techniques like Opinion mining, feature extraction and Naives Bayes classification for review polarity determination. The authors suggest performing both Objective and Subjective analysis of features by considering qualitative and quantitative features of the data respectively.
- Wang et al. (2018)
  - Authors have proposed a solution by implementing Kansei engineering and text mining simultaneously which will help customers in decision making process. It helps to categorize reviews into multiple sections and perform text mining by NLP techniques like Sentence segmentation, Tokenization, and POS tagging.

1.5 Research Questions

- Can the model from Guha Majumder, Dutta Gupta, and Paul (2022) be generalized with:
  - larger volume of products and product types from which to mine data?
  - a sliding scalar multiplier representing the degree to which a product is a "search" (0) or "experience" (1) product?
  - Adding modifiers to review content based upon:
    * Customer / Reviewer reliability and reputation?
    * Review Polarity?
- Can the polarity of reviews be judged accurately by using a Naive Bayes classification model? Hu, Gong, and Guo (2010)
  - What is the impact of different feature extraction methods (e.g., bag-of-words, TF-IDF) on the performance of Naive Bayes classification model? Wang et al. (2018)
- Can products be classified on their degree of being search or experience based by examining product variables such as:
  - Degree of specificity in the product description? (e.g. level of detail, length, numeric values, descriptive values may suggest the product is more search than it is experience-based)
  - Whether the product is offered in brand-new condition only, or offered as new, used, or refurbished? (e.g. refurbished products may be more search products than they are experience products)
  - Which of the 5 senses the product engages? (e.g. engagement of more senses, or engagement of solely specific senses like hearing and vision may suggest more experience-based than search based; examine relationship between search and experience vs. senses engaged)
  - Item rarity (limited production or unique items vs. bulk-produced items)? (e.g. limited production products may be more experience-based than search-based)
- Can newer natrual language processing libraries provide a better fit for Review Content metrics examined by Guha Majumder, Dutta Gupta, and Paul (2022)?
- How does sentiment in customer reviews correlate with customer satisfaction metrics or sales figures for a particular product?
- Can we categorize customer reviews based on customer experience and sentiment?
- Do specific product star ratings tend to incite more reviews, and if so, how does this impact the overall reputation measurement?
- Are specific quality descriptors in text-based reviews (e.g., 'enthusiastic', 'disappointed') strongly associated with certain rating levels, and how does this association affect product reputation?

1.6 Goals / Definition of Success

- Replicate similar results to Guha Majumder, Dutta Gupta, and Paul (2022) with similar product types
- Expound upon Guha Majumder, Dutta Gupta, and Paul (2022) with additional products, including:
  a. Original products from (paper): Digital Music, Video Game, and Grocery Item
  b. Additional products (Amazon and Target): Furniture Items, Clothing Items, Home Appliances, Books, Cosmetics, Cleaning supplies
  c. Additional Proucts (Amazon, Target, BestBuy): Electronics
  d. Verify goodness of fit of original model
- Determing best metrics and/or modifiers for Review Content and Customer Reliability
- Achieving similar or better fit than original paper's modeling; extrapolate to other product types.
- Determining strength of correlation metrics (support, confidence, lift) between Naive Bayes' classifier for review polarity Hu, Gong, and Guo (2010)
  - Integrate with the model and test if Naive Bayes shows strong correlation metrics.
  - Compare and contrast the model with and without incorporation.
- Successful computation of reputation scores for reviewers
  - Check applicability across all sites used for determination of validity within the model.
  - If valid and applicable, execute model against testing data set to ensure it holds.

1.7 Project Schedule / Timeline

Below in Table 1.1, we lay out the major tasks, deliverables, and their respective due dates for this effort.

Table 1.1: Major Project Tasks

Table 1.1

Task	Due Date
Milestone 1 Submission	Feb 26 2024
Product Identification and Selection	Feb 28 2024
Vendor Identification and Selection	Feb 28 2024
Data Collection	Mar 8 2024
Data Cleaning/Pre-Processing	Mar 17 2024
Milestone 2 Submission	Mar 20 2024
Review Classification (Naive Bayes, Kansei)	Mar 27 2024
Product Classification	Mar 27 2024
Reputation Classification	Mar 27 2024
Exploratory Data Analysis	Mar 31 2024
Milestone 3 Submission	Unknown
Model Selection	April 7 2024
Model Testing:	April 11 2024
Complete Final Paper / Milestone 4	April 17 2024

2 Data Collection and Exploration

2.1 Data Collection Overview

Leverage python Selenium, urllib, and Beautiful Soup to scrape data from X products.

- Sought to collect some data from 3 websites - target, amazon, and best buy.
- Where possible, collect same entity from multiple sites

As part of collection, to the greatest extent we are able, we cleaned information *during* the scraping process. We leveraged tools such as the python regular expression library and (*anything else?*) to pull the exact information we sought while scraping. The only possible additional cleaning that may be required after scraping is the handling of unicode characters within product names, product reviews, and so forth.

In terms of simplicity for scraping our data, we will manually identify a list of products. Guha Majumder, Dutta Gupta, and Paul (2022) leveraged solely 3 products from Amazon. Our team seeks to scrape between 20 and 30 products and all their reviews from Target, Amazon, and Best Buy. By doing so, we are greatly increasing the sample size of data compared to the original work performed.

2.2 Data Collection Details

In collecting our data, in order to adhere to the model implemented by Guha Majumder, Dutta Gupta, and Paul (2022), we require the following data points, at a minimum:

Table 2.1

Variable	Data Type
Product Title	string
Product Category*	string
Product Details/Specs	string
Product Cost	float

Table 2.2

Variable	Data Type
Verified Purchase	boolean
Star Rating	float
Review Content	string
Useful Votes	integer

From our collected data, here are some of the calculations we'll need to run for building our models. If time is available, we will run these calculations using Python's NLTK before the completion of this milestone.

Table 2.3

Variable	Data Type
Product Subjectivity	float
Review Length	integer
Sentiment Score	float
Reputation Score	float
Product Type Score	float
Polarity Score	float

In terms of structuring our stored data, we will have a central table and child tables. Since we will seek, in some cases, to gather the *same* product from multiple websites, we must have a structure that identifies:

- The full listing of products, assigned an arbitrary ID
- A listing of specific products we intend to scrape from each of the websites we've identified. Adding an additional ID of [company_name]-[arbitrary_product_id]
- Site-specific product information
- Site-specific review content and metadata

To wrangle the potential amount of data we may collect, we will partition review files into their own files under the following convention: [company_name]-[arbitrary_product_id]-review_content.csv. Using this method will allow us to capture thousands of reviews for multiple products without overrunning GitHub filesize limitations. Following this convention will also allow us to easily script out the integration of all these files for analysis when we begin building and applying our models.

Additionally, the use of assigning products an arbitrary ID that is common between different vendors will allow us to *directly compare* review and product information across multiple vendors simultaneously to gather additional insights. Effectively, if we group products by this identifier, we can see how the product performs overall in multiple e-commerce platforms. Similarly, we can use this information to evaluate and run comparisons and tests on individual products, given the treatment of their offering on different platforms. This can help generate even more insight before model application.

For instance, we may be able to explore questions like:

- Is the price of a product higher, given it's offered on Amazon, BestBuy, or Target?
- Is a product's star rating affected by which e-commerce platform is selling it?
- Is there a substantial difference in number of product reviews on one e-commerce platform vs. another?
- Is one e-commerce platform more likely to have input and feedback on reviews (i.e. higher proportion of "this review is helpful" votes to total number of reviews)?
- What is the difference in the level of detail provided in product descriptions (e.g. for the same product) across each e-commerce platform?

Structuring our data properly during the collection process will enable us to explore and answer these questions.

2.3 Collection Procdures?

We wrote code to allow us to (mostly) template out our gathering of information from each website. The general process for each page is similar for data gathering. To alleviate any unnecessary burden on the target websites, we manually identified URLs to the specific products we sought out to gather, and wrote our code to iterate through those URLs and pull the necessary data and features we sought. This hybrid approach saved us time and effort.

- Gathering from Target (All products)
  - Target has dynamic content on their webpages. We used Python Selenium to navigate to product pages and automate the selection of items needed to expand sections to reveal additional data. We also automated the process of expanding out all reviews so as to iterate through and parse the content of every review for each product in question. We extracted the fields listed above (reference here) to store in our records tables.
- Gathering from Amazon (All Products)
- Gathering from BestBuy (Electronic Products, Furniture Item(s)? - no grocery or clothing)

2.4 Visualizations

- We require a minimum of 10 unique EDA plots for this milestone. We've outlined some of the below but need our data in order and unified prior to development.
  - Scatter Plots
  - Bar Plots
  - Box Plots
  - Violin Plots (e.g. same product, two different websites)
  - review length
    - vs star rating
    - vs sentiment
  - Product description / detail length
    - Potentially explore "specificity" classifier
  - Correlation analyses and linear regressions
  - Heatmaps
  - Tukey test visuals
- Inter-Website Comparison of Product Reviews
  - Same Product
    - Clustering?
    - Distances?
  - All Products
  - Inspect the following, visually:
    - Product Ratings

        * Customer Sentiments **try to score before plotting & turn-in**
        * Review Polarity **try to score and store before plotting**
        * Naive Bayes Classifier
        * Reliability estimates
        * Product description subjectivity scores **try to score and store before turn-in**
        * Average / Spread of number of ratings per product, **try to score and store before turn-in**
        * Average/Spread of Useful Votes per Product Review, **try to score and store before turn-in**
        * Inspection of Data and / or Scoring using Kansei method.

- Will need to take note on if / how these variables conform to some form of statistical distribution (uniform, normal, exponential, etc)

2.5 Data Before / After

-

2.6 Insights from Collection and EDA

-

3 Models Implemented

Here's where we'll list out our models

3.1 Examined Models from Literature Research

- Perceived Usefulness of Product Reviews
  - Modeling for search vs. experience goods as "mitigations" on reviews, examining their impact on perceived review usefulness
- Wealth of Data Paper on Customer Evaulation
  - Examination of 'Vine' customers from Amazon
  - Don't think that BestBuy or Target have the same.
  - Scope of this would be limited to Amazon, combining research from Perceived Usefulness paper (i.e. their modeling) with a customer "trustworthiness" score for Amazon Vine customers
    * May have challenges identifying products with Vine customers
    * Should also have a look at "verified purchase" customers, too.
  - Weaknesses (from authors)
    * Their NLP instance wasn't well trained for fully accurate
    * Their weighting system was manually (i.e. arbitrarily) set as opposed to having a labeled dataset
    * Can we fix these problems? If so - what data is needed?

3.2 Potential Model for Search vs. Experience Weight (scale of 0-1)

3.2.1 Jaccard Similarity Measure

https://www.sciencedirect.com/science/article/pii/S1567422318300450

- Can be used to produce a similarity between two items on a scale of 0 to 1
- May be able to use this for evaluation of a novel item against a pure-search good vs. a pure-experience good.
  - May require additional calculation / computation between these two values - maybe its arithmetic or harmonic mean between search and experience
  - Using that, we could potentially produce a weight.
  - Authors models
    * Model 1: Perceived Usefulness $= \beta_0 + \beta_1 \cdot \text{Review Content} + \beta_2 \cdot \text{Review Length} + \beta_3 \cdot \text{Star Rating} + \beta_4 \cdot \text{Total Votes Received} + \epsilon_1$

- Model 2 (product type as a moderator): Perceived Usefulness $= \beta_0 + \beta_1 \cdot$ Review Content $+ \beta_2 \cdot$ Review Length $+ \beta_3 \cdot$ Star Rating $+ \beta_4 \cdot$ Total Votes Received $+ \beta_5 \cdot$ Digital Music $+ \beta_6 \cdot$ Video Game $+ \beta_7 \cdot$ Review Content $\cdot$ Digital Music $+ \beta_8 \cdot$ Review Content $\cdot$ Video Game $+ \epsilon_2$
- Our proposition #1 (may require some modification)

  Perceived Usefulness $= \beta_0 + \beta_1 \cdot$Review Content$+ \beta_2 \cdot$Review Length$+ \beta_3 \cdot$Star Rating$+ \beta_4 \cdot$Total Votes Recei

or

Perceived Usefulness $= \beta_0 + \gamma \cdot \beta_1 \cdot$Review Content$+ \beta_2 \cdot$Review Length$+ \beta_3 \cdot$Star Rating$+ \beta_4 \cdot$Total Votes Received

- Where γ is the Jaccard Similarity Score between a given product and elements we are identifying as "pure" experience and "pure" search good.
- Operationalizaton of Jaccard Similarity score Variable (i.e. inputs)
  - Search good: those with attributes that can be evaluated prior to purchase or consumption. Consumers rely on prior experience, direct product inspection and other information search activities to locate information that assists in the evaluation process. Most products fall into the search goods category (e.g. clothing, office stationery, home furnishings).
  - Number of measurement specifications?
  - comment/review information?
  - What else could we gather that could be considered a "universal" tangible or intangible feature from a product online? They need to be applicable to both search and experience goods.
- Experience good: those that can be accurately evaluated only after the product has been purchased and experienced. Many personal services fall into this category (e.g. restaurant, hairdresser, beauty salon, theme park, travel, holiday).
  - subjective descriptiveness vs.
  - comment/review information?
- Are there other methods aside from Jaccard Similarity?

3.3 Model Comparison

4 Conclusion

Here's our conclusions section

References

Guha Majumder, Madhumita, Sangita Dutta Gupta, and Justin Paul. 2022. "Perceived Usefulness of Online Customer Reviews: A Review Mining Approach Using Machine Learning & Exploratory Data Analysis." *Journal of Business Research* 150 (November): 147–64. https://doi.org/10.1016/j.jbusres.2022.06.012.

Hu, Weishu, Zhiguo Gong, and Jingzhi Guo. 2010. "Mining Product Features from Online Reviews." *2010 IEEE 7th International Conference on E-Business Engineering*, November. https://doi.org/10.1109/icebe.2010.51.

Rajeev, P Venkata, and V Smrithi Rekha. 2015. "Recommending Products to Customers Using Opinion Mining of Online Product Reviews and Features." *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*, March. https://doi.org/10.1109/iccpct.2015.7159433.

Wang, W. M., Z. Li, Z. G. Tian, J. W. Wang, and M. N. Cheng. 2018. "Extracting and Summarizing Affective Features and Responses from Online Product Descriptions and Reviews: A Kansei Text Mining Approach." *Engineering Applications of Artificial Intelligence* 73 (August): 149–62. https://doi.org/10.1016/j.engappai.2018.05.005.