Final Report: PureView AI, An Unbiased NLP Tool For Customer Sentiment

Team 184

David Llanso / Carson Stone / Jeffrey Triemer / Oswaldo Ceballos / Caitlyn McCoole

Introduction and Motivation

The motivation for our project stems from the increasing concerns surrounding the potential biases in product review analysis on major e-commerce platforms like Amazon. With Amazon's market dominance, the influence of biased sentiment and topic modeling could significantly impact consumer behavior and product visibility. Recognizing the need for transparency and fairness, our team aims to develop an unbiased NLP tool that offers consumers an independent and reliable source for understanding product sentiments. By empowering users with clear and objective insights, we hope to promote trust and accountability in online reviews, ultimately benefiting both shoppers and the broader marketplace.

Problem Definition

In 2017, The Yale Law Journal, in an article by the current FTC Chair Lina Khan, highlighted the underappreciated antitrust risks posed by predatory pricing and cross-business integration. Recognizing that Amazon's market share—46% of online shopping at the time—demands scrutiny, we explored how sentiment and topic modeling of reviews could contribute to their competitive advantage. To address this, we developed an independent NLP tool that audits Amazon's modeling practices and offers transparent insights into product reviews. Our Tableau dashboard incorporates sentiment and topic modeling, allowing users to filter by product attributes, review sentiment, and topics, using open-source tools for clarity and trust. Despite challenges with data volume and potential biases, our solution provides an accessible and reliable third-party analysis of customer sentiment, independent of Amazon's influence.

Literature Survey

Conducting a thorough literature review is crucial for understanding existing methodologies and informing the development of our project. Birjali et al [1] provides a comprehensive comparison of traditional and modern sentiment analysis methods, highlighting model selection reasoning. Despite its broad, non-specific focus, it offers valuable objectivity for our research. Distante et al [2] covers topic modeling, while Jones K.S. et al [3] explores the history of NLP, both informing our approach to modeling and understanding NLP's evolution. Haque, T.U. et al [4], Rathor, A.S. et al [4], and Rashid, A. et al [5] specifically analyze Amazon reviews, offering methodologies we can refine. Aldunate et al [8] introduces BERT for deep-learning text classification, relevant to our model's development. Kiran [11] and Sharma et al [13] extend NLP workflows with product-based recommendations, insights we plan to enhance. Park et al [9], Rao [12], and Hasan et al [16] address NLP challenges with slang and jargon, using varied techniques that may adapt well to structured Amazon reviews.

Proposed Methods

Setup

Our task requires that we use a lot of computational power, and a way to share a codebase and environment for development. We used Google cloud platform (GCP) to create a shared project, input our data into a storage bucket in that project, and then create a Vertex AI workbench instance with some starter code to read from the bucket. We also created an ipykernel to ensure that the team was using the same packages for development, with some readme instructions for instance and environment setup from the GitHub. From here, the team was ready to begin development.

Preprocessing

There was a substantial amount of preprocessing and computation needed to get the data exactly as we needed it. Our goal was initially to just use the reviews dataset, but we quickly learned we needed an adjacent dataset, product metadata, to actually see detailed information about the products in reviews. Since the review dataset was already over 10GB to begin with, bringing in the metadata was not simple. One innovation that allowed us to work with such a large volume of data, was processing our preprocessing code in chunks, instead of trying to do it all at once.

There were a handful of cleaning tasks that needed to be completed before we started EDA. One was tokenization. We took the review text column, and utilized the NLTK package to tokenize, lemmatize, and remove stop words. Another key innovation in this step was keeping reviews that have been rated as helpful, and ones for verified purchases. These indicators helped us filter out low quality reviews, trolls, fake reviews and/or spam that may otherwise cloud the results.

Topic Modeling/EDA

Topic modeling made up a portion of our final dashboard and is a key piece of understanding the overall picture of sentiment. Topic modeling refers to the idea of taking text, or some other form of speech, and extracting a few meaningful topics or themes that most closely associate. We achieved this by expanding on our preprocessing step and configuring word clouds for each product with python's NLTK package. Our final dashboard includes words clouds where users are able to see a handful of themes for each product and on a category level. This helps give context behind the sentiment scoring and gives people a quick glance at the 'why' behind the score. On top of implementing methods of topic modeling, we also visualized our data in matplotlib to understand our data better. Some of these visualizations are part of our final dashboard.

Sentiment Modeling

One innovation in our approach to sentiment scoring involved testing multiple types of sentiment analysis algorithms spanning from simple text vectorization to deep learning. We used VADER, BERT, and Bag-of-words to understand and predict sentiment scores based on customer reviews. These 3 algorithms presented a range of varying complexity, interpretability, and robustness. We then measured the accuracy of these classification models with chi-squares up against the star ratings of the product and compared other things like cost and complexity to choose the best one.

User Interfaces:

The tableau dashboard has a summary tab and a detail tab. The summary tab shows an aggregation of the data at a high level, with a large selection of filters for users to utilize in searching for their specific use case. This gives users some additional opportunities for analysis, like an overview of highest performing topics, products or product types in terms of sentiment. The reason we do not offer a direct line to the product details at a lower level upfront is so that users are guided to filter the data down to a lower subset before loading any lower-level details. This provides a more natural experience, particularly in terms of managing Tableau's performance. Loading tables in

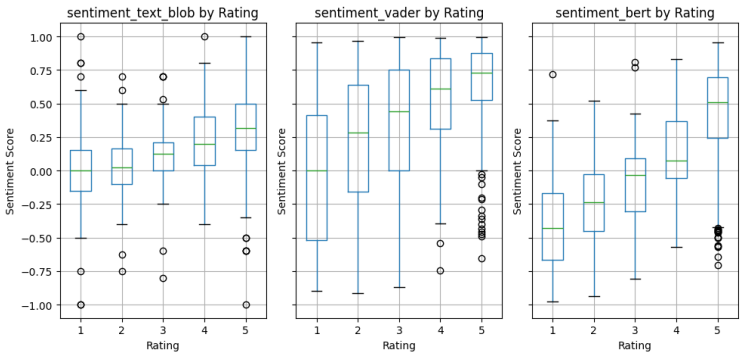
Tableau takes a lot of computation, so we leveraged this technique to make users filter down ahead of pulling any of the lower-level table information. The detail tab gives a line by line look at each review, their rating, their comment, the topic, the sentiment and any other additional product information that might be relevant to a shopper. We also provide a high-level summary of rating and sentiment towards the top of the dashboard so that users don't feel lost in the weeds of individual customer reviews.

Experiments and Evaluation

Visual analytics scalability experiment

Description and goals- We will have a chokepoint of our tableau dashboard housing an extract as large as 28gbs. This is no small feat, and we will need a way to validate that we can display data this large, specifically in a tableau public environment, without harming the end user experience so significantly that we risk losing possible users due to performance issues. **Design**- The team will need a sample 10gb dataset to extract within tableau with like dashboard filters/actions to see whether or not a dashboard of that size is viable. If not, there may be some filtering we can do to the front end of the tool or other filters to cut down on the size of our data. The success criteria will be if the tableau dashboard can load in tableau public within 30 seconds and the actions and filters do not take longer than 30 seconds. The question at hand was how we would scale or store this data to be used in a Tableau dashboard, much less in the modeling phase. **Results-** We tried reading in the data in chunks, but this seemed to constantly crash our google cloud instance. We ultimately concluded to use a slice of the data, instead of the entire dataset, for the purposes of building a working model, and putting together a cohesive dashboard. In future iterations or updates, it would be a simpler process, as we could load in just the new review data in batches, instead of all cataloged reviews at one time.

Sentiment accuracy experiment



Description and goals- The presentation of our project's sentiment analysis will only be as good as its sentiment accuracy. We need to validate the sentiment our model is providing so that we can have confidence we are providing our end user with legitimate insights that they can apply in their shopping process. **Design -** To avoid the manual effort, we could also leverage the customer reviews provided to validate whether the sentiment the model assigned was accurate. For example, if a review gives a product 5 stars, the reviews text should be largely positive, perhaps with some specific critiques. Sentiment modeling is a fantastic way to make qualitative data quantitative. This transformation of the data is its main value; however, this provides a unique challenge in validating the model's performance. For many sentiment models, their quantitative output must somehow to be compared to its qualitative input. This could mean a very manual effort to read many reviews to assess the model's accuracy. In our case, we were fortunate to have a proxy for review sentiment in a user's 1-5 rating of their given Amazon product, so we plotted each of 3 sentiment models' outputs in boxplots by rating. This easily conveyed how similar the models were to the users' rating. We used this information to

select our model and validate its effectiveness. **Results-** We found that VADER performed the best in output and computational efficiency. Observing the boxplot, it seems like sentiment responded as expected with users' ratings. BERT arguably may have had better results, but we found that BERT was computationally more expensive than VADER, for results that weren't significantly better. We concluded that VADER was the best tool for our use case, due to its relationship with user ratings and its computational advantages in comparison with BERT.

Topic accuracy experiment

Description and goals- The value of our tool is primarily in the sentiment analysis by product, with the topic summary as a second. We plan to offer the users some estimate idea of what specific topics about the product customers had negative or positive experiences with. To offer this kind of product effectively, we need to have confidence that the topics we highlight or truly important to the customer's experience as well as the current shopper. We also need to know that the sentiment can be effectively grouped at that topic level. **Design**- We used LDA (Linear Discriminant Analysis) model to extract the top 3 topics from each review using the TF-IDF framework. These topics would represent the most used topics, as well as the rarest topics across all reviews, to determine the 3 most significant. If there were not 3 significant topics, the others would be left blank. This was more common than we anticipated as many reviews only contained a few words, especially after removing stop words. **Results-** This model allowed us to build word clouds in our Tableau dashboard and shows users the most common topics for the products and categories they've selected.

Conclusions and Discussion

Our goal was to take a raw text reviews dataset spanning multiple categories as an input for sentiment and topic models that we would set up and train, and to use the output from these models to derive insights for the average e-commerce customer. At the onset, we encountered challenges with importing and processing the full dataset into the shared storage of our cloud computing environment, leading us to realize that we would need to prepare the data in more a condensed and efficient manner rather than bulk loading every row of every column. This was ultimately achieved through trial and error until we determined the features and scale of the dataset that would be sufficient for our experiment.

The Linear Discriminant Analysis topic model was implemented to determine the 3 most significant topics per product based on the product reviews from a combination of the most common and rare topics. Using the output data, the first 2 columns of the 3 containing the most significant topics were used to create word clouds, one for each column. From these, we were able to successfully visualize topic words in comparison to each other based on their prevalence in the output data, and this can be analyzed on a deeper level using the product category filter and sentiment score slider on the dashboard. From inspecting the dashboard results with different sentiment scores, the word clouds appeared to correlate with the scoring.

The boxplot sentiment models were implemented to compare the distributions of product ratings with sentiment scores. We determined that VADER was the most efficient and precise method at producing sentiment ratings that correlated with product ratings based on the boxplot outputs.

In the end, you can see in our GitHub that our group learned to make effective sentiment and topic modeling using best practices such as git for version control, Google cloud platform for google cloud storage, Vertex AI computing, and environment management via GCP Vertex AI instances. In

reviewing our GitHub, you will see we include directions for how to setup use in GCP, as well as local execution of our script, which includes detailed documentation on the tool's functions and objectives. This all creates a functioning sentiment and topic modelling technique for Amazon's product reviews that allows users to fully trust the source of their data and pick up our modeling where we left off with little effort.

All team members have contributed a similar amount of effort.

Sources:

Akhilesh Kumar Sharma, Bhavna Bajpai, Rachit Adhvaryu, Suthar Dhruvi Pankajkumar, Prajapati Parthkumar Gordhanbhai, Atul Kumar (2023). An Efficient Approach of Product Recommendation System using NLP Technique. Elsevier. (3730- 3743). https://doi.org/10.1016/j.matpr.2021.07.371.

Ángeles Aldunate, Sebastián Maldonado, Carla Vairetti, Guillermo Armelini (December 15, 2022). Understanding customer satisfaction via deep learning and natural language processing. Elsevier. https://www.sciencedirect.com/science/article/abs/pii/S0957417422014397

Birjali, M., Kasri, M., & Beni-Hssane, A. (2021, May 14). A comprehensive survey on sentiment analysis: Approaches, challenges and Trends. Elsevier. https://www.sciencedirect.com/science/article/abs/pii/S095070512100397X

Distante, E. (2022, October 20). Bertopic: Topic modeling as you have never seen it before. Medium. https://medium.com/data-reply-it-datatech/bertopic-topic-modeling-as-you-have-never-seen-it-before-abb48bbab2b2

Fang, X., Zhan, J. (June 16, 2015). Sentiment analysis using product review data. Journal of Big Data. https://doi.org/10.1186/s40537-015-0015-2

Hasan, Ali, Sana Moin, Ahmad Karim, and Shahaboddin Shamshirband. 2018. "Machine Learning-Based Sentiment Analysis for Twitter Accounts" Mathematical and Computational 23(1). https://doi.org/10.3390/mca23010011

Haque, T. U., Saber, N. N., & Shah, F. M. (May, 2018). Sentiment analysis on large scale Amazon product reviews. 2018 IEEE International Conference on Innovative Research and Development (ICIRD). https://doi.org/10.1109/icird.2018.8376299

Jones, K. S. (1994, January 1). Natural language processing: A historical review. SpringerLink. https://link.springer.com/chapter/10.1007/978-0-585-35958-8_1

Khan M., Lina (January, 2017). Amazon's Antitrust Paradox. The Yale Law Journal. 564- 907.

https://www.yalelawjournal.org/note/amazons-antitrust-paradox

M. V. K. Kiran, R. E. Vinodhini, R. Archanaa and K. Vimalkumar (January, 2017). User specific product recommendation and rating system by performing sentiment analysis on product reviews. International Conference on Advanced Computing and Communication Systems (ICACCS), 1-5. doi: 10.1109/ICACCS.2017.8014640

Madhumita Guha Majumder, Sangita Dutta Gupta, Justin Paul (November, 2022). Perceived usefulness of online customer reviews: A review mining approach using machine learning & exploratory data analysis. Journal of Business Research. 150 (147-164), https://doi.org/10.1016/j.jbusres.2022.06.012.

Nadia Malik, Muhammad Bilal (July 19, 2024). Natural language processing for analyzing online customer reviews: a survey, taxonomy, and open research challenges. PeerJ Computer Science. https://peerj.com/articles/cs-2203/

Noori, B. (May 6, 2021). Classification of Customer Reviews Using Machine Learning Algorithms. Applied Artificial Intelligence, 35(8), 567–588. https://doi.org/10.1080/08839514.2021.1922843

Rao, K. Yogeswara, G. S. N. Murthy, and S. Adinarayana (July, 2017). Product recommendation system from users reviews using sentiment analysis. International

Journal of Computer Applications. https://www.researchgate.net/profile/Adinarayana-Salina/publication/318493578

Rashid, A., & Huang, C. (May 2, 2021). Sentiment Analysis on Consumer Reviews of Amazon Products [Review of Sentiment Analysis on Consumer Reviews of Amazon Products]. Research Gate; International Journal of Computer Theory and

Engineering. https://www.researchgate.net/profile/Ching-Yu-Huang/publication/355706880_Sentiment_Analysis_on_Consumer_Reviews_of_Amazon_Products/links/6254681eef013420666a6fbe/Sentiment-Analysis-on-Consumer-Reviews-of-Amazon-Products.pdf

Rathor, A. S., Agarwal, A., & Dimri, P. (2018). Comparative Study of Machine Learning Approaches for Amazon Reviews. Procedia Computer Science, 132, 1552–1561. https://doi.org/10.1016/j.procs.2018.05.119

Sunghong Park, Junhee Cho, Kanghee Park, Hyunjung Shin (September, 2021). Customer sentiment analysis with more sensibility (2021). Elsevier. https://www.sciencedirect.com/science/article/abs/pii/S0952197621002049

Wallace, Alicia (May 21, 2024). High Inflation Made Finances Worse for 65% of Americans Last Year. CNN. https://www.cnn.com/2024/05/21/economy/economic-wellbeing- 2023-

inflation/index.html

Wells, John R., Benjamin Weinstock, Gabriel Ellsworth, and Galen Danskin (August, 2021). Amazon.com, 2021. Harvard Business School Case, 716-402. https://www.hbs.edu/faculty/Pages/item.aspx?num=49324