Topic Modeling in Financial App Negative Reviews

Group 12
Matt Drezdzon
Grace Liu
Kevin Chien
Yixuan (Amy) Ding

Natural Language Processing Final Project

Abstract

As trading apps have become increasingly prevalent in the financial sector, attracting a wide range of users, their user experience (UX) has become crucial in retaining users and maintaining a competitive edge. The present study applies natural language processing (NLP) techniques to analyze user reviews from four major trading apps — Fidelity Investments, Robinhood, Coinbase, and Moomoo — to identify sentiment trends and UX pain points. Our methodology involves a multi-step pipeline: scraping App Store reviews, text preprocessing, sentiment analysis using the VADER lexicon-based model, and topic modeling with LDA on negative reviews. We also perform TF-IDF keyword extraction and manual validation to ensure interpretability. The analysis reveals app-specific issues, such as account setup issues and a lack of trade efficiency with Fidelity Investments, withdrawal delays and market distrust with Robinhood, poor customer support and lack of fee transparency with Coinbase, and user interface and onboarding issues with Moomoo; these issues have real consequences for user trust, financial outcomes, and app ratings. We not only offer personalized recommendations to improve the UX for all four apps, but also demonstrate the value of NLP-driven sentiment and topic analysis for improving customer satisfaction and identifying competitive gaps.

Introduction

With the digitization trend, more and more people are exposed to a wider breadth of information. Organizations are also conducting transformations to adopt this change to accept more diverse customers and lower the barriers to new user entry. While previously only professional traders would use traditional investment platforms, and there were hidden restrictions for ordinary people, now, with emerging mobile trading applications and higher market accessibility, more everyday users can leverage investment platforms to make their financial decisions. However, with more new users entering the market, financial service organizations also face new challenges in user experience (UX) issues, which would directly impact the investors' success, investors' loyalty rate, and companies' competitiveness.

Since UX issues in the financial service industry can lead to significant adverse impacts on companies and users, ensuring an intuitive UX is crucial. Unlike design in non-financial mobile applications, where poor UX only causes frustration and does not necessarily have substantial consequences, poor design in financial apps can result in monetary losses, mistaken trades, and erosion of user trust. With the impact of the recent COVID-19 pandemic, the competitive landscape of trading applications has intensified. According to a 2021 study by the FINRA Investor Education Foundation and NORC at the University of Chicago, 57% of investors opened new investment accounts in 2020, and 66% were first-time investors using digital platforms. With more younger, lower-income, and diverse users, financial organizations' UX quality becomes a differentiator, especially for those with limited financial knowledge.

In recent years, more and more research has been done in fintech and user experience. A study by Xu et al. (2024) identifies user interface (UI) design as a pivotal component in fintech development. For companies integrating AI to improve their UX design, there is a 41% increase in daily active users. Similarly, Signicat (2022) reports that 68% of new users stop onboarding with financial services when they face a poor or cumbersome user experience, highlighting the critical role of intuitive design in customer retention. Moreover, a study by Chaudhry and Chinmay (2021) finds that when trading app pages are unclear or poorly structured, users will decrease their trade accuracy and delay their execution, which may increase misinterpretation for key financial signals as well as their cognitive strain, thus potentially leading to suboptimal investment decisions and financial losses.

At the same time, in the domain of user feedback analysis, several studies have already used natural language processing (NLP) techniques to extract insights from app reviews. Guzman and Maalej (2014) first applied topic modeling and sentiment analysis to App Store reviews and eventually realized a precision score of 0.59 and a recall of 0.51, which shows the methods' efficacy in identifying recurrent themes in user feedback. Building on this work, Shah et al. (2024) used large language models (LLMs) to elevate the traditional model, which can automatically extract fine-grained app features and associated sentiments from user reviews.

Researchers have already realized some advanced NLP applications, but a gap exists for systematically identifying and prioritizing UX improvements in trading applications. Most current research on financial apps relies only on survey data or user test results. There is also seldom a comparative analysis across different trading platforms, which may limit our understanding of industry-wide patterns and competitive differentiators.

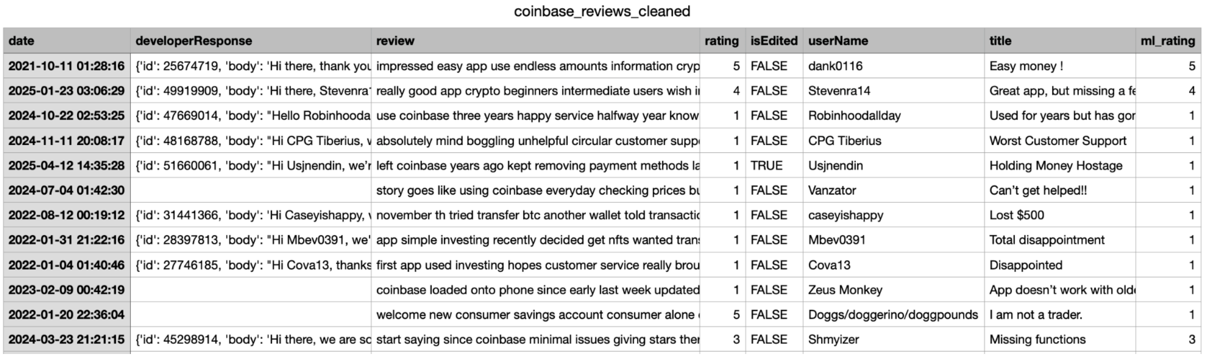
Our research addresses these gaps by applying advanced NLP techniques to App Store reviews of four distinct trading applications: Fidelity Investments, Robinhood, Coinbase, and Moomoo. Our app selection aims to cover the diversity across the financial services market. Fidelity is a more traditional and comprehensive investment platform that offers a wide range of financial services. Robinhood is known for its commission-free trading model and simplified user interface. Coinbase is a major cryptocurrency exchange platform widely used for digital asset transactions and portfolio management. Moomoo is a new generation mobile-based trading platform with rich analytical tools and visualizations. Together, these platforms allow us to explore UX challenges across traditional, crypto, and next-gen trading apps. By applying machine learning models, sentiment analysis (VADER), and topic modeling (LDA) to these apps' negative reviews, we aim to uncover the platform-specific UX issues and thus enable companies to achieve product improvements, competitive advantage, and reputation management.

Data

Our study extracted our dataset from the Apple App Store across four trading applications: Fidelity, Robinhood, Coinbase, and Moomoo. We used an App Store scraper package to collect a random sample of 1,000 reviews per app across the past five years (2021-2025), which provided us with a comprehensive view of user experiences over time. Since Fidelity, Robinhood, and Coinbase have longer operational histories and thus enough historical data, we collected 200 randomly sampled reviews from each year to ensure temporal coverage. Since Moomoo is a relatively new app and has not been live for the same number of years as the other apps, we collected 1,000 randomly sampled reviews from 2022 to 2025, with 250 from each year. This sampling strategy also ensures the coverage of historical and recent user experiences, including reviews related to platform updates or market events.

Each review in our dataset contains several key pieces of information like review title, review content, star ratings (1-5 scale), user name, review date, and developer's responses (if available). We also matched a machine learning predicted rating score and sentiment analysis results to the dataset later in our process. The dataset was stored in CSV format. We used Python for further analysis, including preprocessing, sentiment scoring, and topic modeling.

An example of our dataset is shown below in Figure 1.



coinbase_reviews_cleaned

date	developerResponse	review	rating	isEdited	userName	title	ml_rating
2021-10-11 01:28:16	{'id': 25674719, 'body': 'Hi there, thank you	impressed easy app use endless amounts information cryp	5	FALSE	dank0116	Easy money !	5
2025-01-23 03:06:29	{'id': 49919909, 'body': 'Hi there, Stevenra1	really good app crypto beginners intermediate users wish ii	4	FALSE	Stevenra14	Great app, but missing a fe	4
2024-10-22 02:53:25	{'id': 47669014, 'body': "Hello Robinhooda	use coinbase three years happy service halfway year know	1	FALSE	Robinhoodallday	Used for years but has gor	1
2024-11-11 20:08:17	{'id': 48168788, 'body': "Hi CPG Tiberius, v	absolutely mind boggling unhelpful circular customer supp	1	FALSE	CPG Tiberius	Worst Customer Support	1
2025-04-12 14:35:28	{'id': 51660061, 'body': "Hi Usjnendin, we'r	left coinbase years ago kept removing payment methods la	1	TRUE	Usjnendin	Holding Money Hostage	1
2024-07-04 01:42:30		story goes like using coinbase everyday checking prices bu	1	FALSE	Vanzator	Can't get helped!!	1
2022-08-12 00:19:12	{'id': 31441366, 'body': 'Hi Caseyishappy, v	november th tried transfer btc another wallet told transactio	1	FALSE	caseyishappy	Lost $500	1
2022-01-31 21:22:16	{'id': 28397813, 'body': "Hi Mbev0391, we'	app simple investing recently decided get nfts wanted trans	1	FALSE	Mbev0391	Total disappointment	1
2022-01-04 01:40:46	{'id': 27746185, 'body': "Hi Cova13, thanks	first app used investing hopes customer service really brou	1	FALSE	Cova13	Disappointed	1
2023-02-09 00:42:19		coinbase loaded onto phone since early last week updated	1	FALSE	Zeus Monkey	App doesn't work with olde	1
2022-01-20 22:36:04		welcome new consumer savings account consumer alone (	5	FALSE	Doggs/doggerino/doggpounds	I am not a trader.	1
2024-03-23 21:21:15	{'id': 45298914, 'body': 'Hi there, we are sc	start saying since coinbase minimal issues giving stars ther	3	FALSE	Shmyizer	Missing functions	3

Figure 1: Sample Dataset from Coinbase

Methodology

We conducted a multi-stage analysis of trading app user reviews to extract key insights from user sentiment and identify pain points. Our methodology followed a three-step process: preprocessing, sentiment analysis, and topic modeling. First, all reviews were preprocessed for the topic modeling step. We standardized all text by lowercasing all content, removing punctuation (except apostrophes), emojis, and stopwords. We opted not to perform stemming or lemmatization as we wanted to preserve the contextual integrity of the user reviews for our sentiment and topic analyses. Next, for sentiment analysis, we utilized the Valence Aware Dictionary and Sentiment Reasoner (VADER) to classify the reviews as positive, neutral, or negative. We manually spot-checked reviews and their original ratings with the VADER sentiment labels to validate VADER's accuracy. If the label outputs did not align with the original reviews, we adjusted the sentiment threshold for better alignment accordingly for each of our apps. We applied TF-IDF to sort out high-weight keywords to interpret sentiment trends further and analyzed sentiment distribution across different app experiences. We also developed our machine learning models to gain more context into the relationship between the star rating each user assigned to their review and the sentiment of the review text. Finally, we used Latent Dirichlet Allocation (LDA) topic modeling exclusively on negative reviews to uncover recurring topics in negative reviews. We validated LDA's topic outputs by manually spot-checking randomly selected reviews to ensure interpretability and alignment with user concerns.

The architecture of the project follows the flowchart in Figure 2 below.

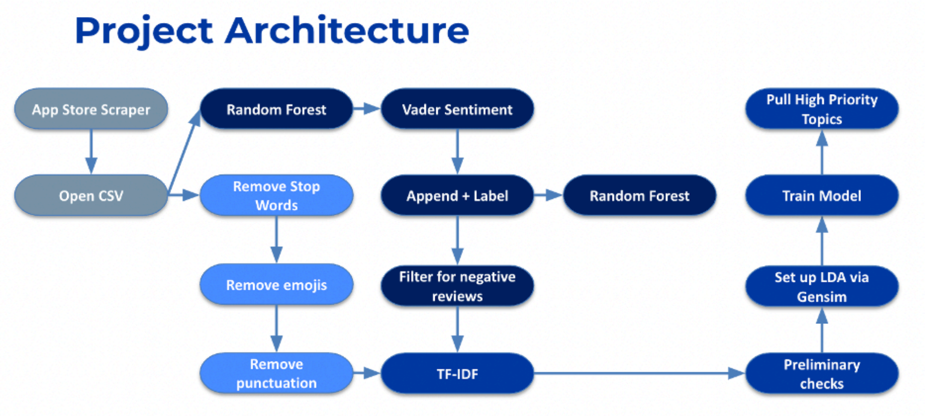


Figure 2: Project Architecture Flowchart

We began the project by connecting to the App Store Scraper package, which allows us to download raw review data for any app available on the Apple App Store. We can pass in parameters such as the number of reviews we want and the date range we are interested in. Each review contains rich, user-generated organic language that reflects genuine user sentiment and their experiences with the app. Scraping this data ensures we work with directly relevant, real-world text that can impact product enhancements to improve UX. Once collected, we saved the reviews in a CSV file, which was opened in our Python environment with pandas. With our ultimate goal being topic modeling, we conducted preprocessing for the topic modeling step in another column by lowercasing all text and removing stop words, emojis, and punctuation except apostrophes (to preserve situations such as contractions and possessives) to reduce noise in our corpora. Stop word removal prevents the topics we pull from including noise from uninformative words, while emoji and punctuation removal help minimize any possible distortions in vector representations or model training. This cleaning prepares our corpora for more effective vectorization and downstream topic modeling.

For our topic modeling of negative reviews to be reliable, it is essential to isolate the truly negative reviews; this is the most critical piece of our analysis. For exploratory analysis, we explored whether the review text matches the rating. We trained a supervised Random Forest classifier to predict the star rating of each review, using only the raw review text after converting to TF-IDF vectors. TF-IDF in this scenario increases the weighting for more unique, contextual words, which allows the model to focus on review-specific language that may indicate a strong sentiment in either direction, giving us more meaningful feature splits, avoiding the risk of overfitting, and improving the interpretability of the results. Furthermore, Random Forest was chosen for this project due to its resiliency to noisy features (like praises and complaints together), useful feature selection through tree splits (without needing feature selection or dimension reduction), nonlinear modeling ability, and minimal processing needs (Egger & Yu, 2022). This step aimed to determine whether there was a strong linguistic signal between review text and star ratings; if the Random Forest classifier had strong performance, it would suggest some alignment between written language and numerical rankings, while weak performance would indicate a need for more specialized sentiment tools. However, due to the complexity of user-generated reviews, the range of human emotions, and even instances of incorrectly entering the desired rating, analyzing review sentiment is much more complex than this. We found a weak alignment between ratings and the sentiment users express in the text. As such, the preliminary Random Forest model confirmed the limitations of relying on star ratings to capture user sentiment, motivating our transition to more sophisticated sentiment analyses to determine which reviews are negative.

Given that our exploratory analysis with the Random Forest classifier did not indicate a strong relationship between reviews and their star ratings, we next turned to VADER to directly assess the sentiment of each review, assigning the text scores that represent the negativity, neutrality, or positivity of the text. While users sometimes give high ratings despite having multiple complaints or low ratings despite using predominantly positive language, VADER is a consistent, lexicon-based measure of the textual sentiment, utterly independent of the star rating, making it much more reliable for pinpointing truly negative reviews. We appended VADER to each of our reviews and classified the review as neutral if the compound score was between -0.05 and 0.05 and positive or negative if the score was greater than 0.05 or less than 0.05, respectively.

Repeating our Random Forest classifier check with additional features such as the VADER score and creating a compound score based on positive and negative word lists (Hu & Liu, 2004), we now see improved performance and are confident in our sentiment labeling. We then filtered our dataset for negative reviews only, as these are typically the most detailed reviews and reveal important feedback about each app; negativity bias has repeatedly demonstrated that experiences of a negative nature tend to have a greater effect on one's emotions and opinions of a particular product than positive experiences (Isnan et al., 2023). This also sets up our model to maximize the relevance of the topics we will discover, allowing developers to better prioritize the most critical areas to work on for app improvement; positive reviews often are more generic and do not provide the same diagnostic context as negative reviews.

Beginning the topic modeling phase, we created a TF-IDF vector on the filtered negative review dataset, converting each review into a weighted feature vector with higher weighting to unique, meaningful words. This enhances the distinctiveness of the topics we will find by emphasizing the high-content vocabulary in the reviews. Although feature reduction methods such as principal component analysis (PCA) and chi-squared can mitigate sparsity and reduce noise in high-dimensional data like App Store reviews, we retained the full vector to preserve rare but contextually critical terms. For example, words like “locked”, “glitch”, or “margin” may not be frequent or discriminative, but are essential in that they capture domain-specific issues that are central to user pain points;

further work could investigate trade offs between topic coherence and term coverage by selectively trimming features based on coherence scores or using domain-specific lexicons. Once our vector was created, we conducted preliminary checks such as assessing the sparsity, checking the vocabulary size, and validating that the vector preserved meaningful, negative information. We also created a corpora dictionary of the reviews and a bag-of-words corpus for entry into the model, an essential training requirement for our chosen model type, requiring raw term distributions to match the assumptions of the algorithm and gather more coherent topics.

Our topic model was set up through the gensim library using LDA to identify the dormant topics in our reviews. LDA is a widely used method for unsupervised topic modeling, allowing similar reviews to be grouped together based on shared patterns of word usage, ultimately revealing budding themes that encompass the main areas of user dissatisfaction (Kumar Gupta, 2022). We trained the model iteratively with our dictionary and bag-of-words corpus, optimizing parameters such as the number of topics and passes. We chose four topics per app to balance model performance and a realistic number of priorities for developers to act on feasibly. Once trained, we extracted the most frequent topics and their associated top 10 words from the model. We then translated these into human-readable themes (such as “simplify account verification” or “streamline buy/sell flows”) to develop a list of the top, prioritized recommendations for developers to build based on clarity and topic volume. By gathering these key areas of concern from App Store reviews, this project offers developers a clear product backlog of what to improve in their apps by turning unstructured, messy feedback into structured, data-driven recommendations.

Results

The analysis commenced with a comprehensive examination of the star rating distributions across four prominent financial applications: Fidelity, Robinhood, Coinbase, and Moomoo.

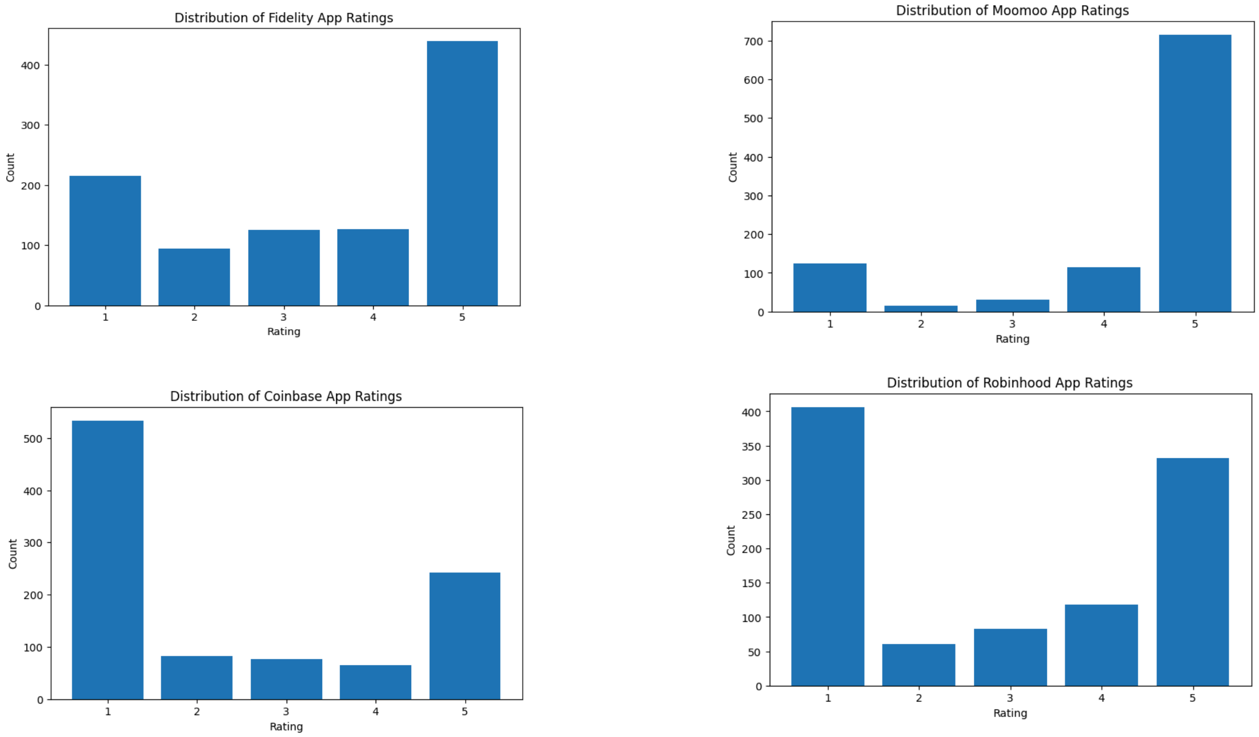


Figure 3: Distribution of Star Ratings Across Apps (User-Provided)

These distributions (Figure 3) illustrate significant variances in user sentiment, highlighting crucial imbalances in rating distributions. Specifically, Robinhood and Coinbase exhibited strongly bimodal distributions dominated by 1-star and 5-star ratings. Robinhood had 406 1-star reviews versus 332 5-star reviews, while Coinbase exhibited an even greater disparity with 533 1-star ratings compared to only 242 5-star ratings. Contrastingly, Moomoo demonstrated a positively skewed distribution with an overwhelming majority (715 out of 1000 reviews) granting a 5-star rating, indicative of overall positive user sentiment. Fidelity showed a relatively balanced distribution, though still skewed positively, with a notable count of 439 5-star ratings compared to 215 at 1-star.

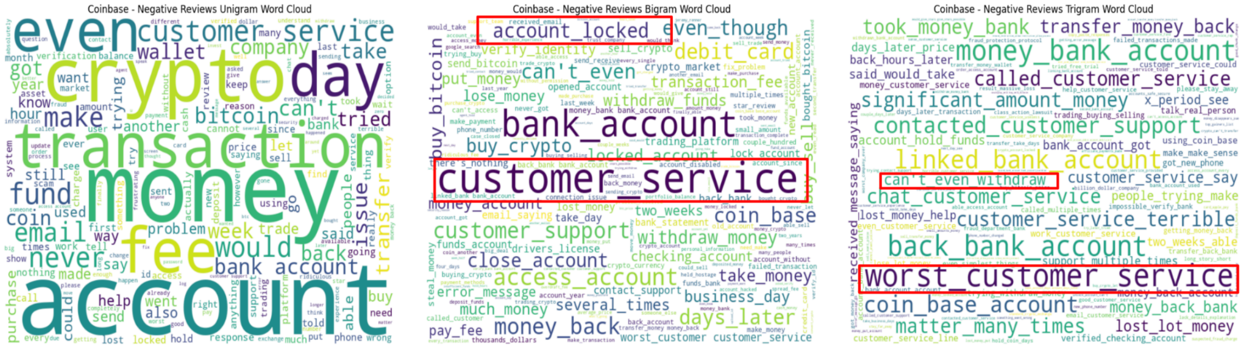


Figure 4: Word Cloud of Most Frequent Unigrams, Bigrams, and Trigrams (Coinbase)

A word cloud analysis provided deeper qualitative insights into user dissatisfaction and praise across the apps. Figure 4 shows an example across unigrams, bigrams, and trigrams for Coinbase. Negative reviews revealed consistent themes such as difficulties with "account access," "money withdrawal," and significant frustrations regarding "customer service." Coinbase negative reviews, in particular, emphasized severe issues with "account locked," "bank account" complications, and consistently "worst customer service," as highlighted clearly from unigram to trigram analyses. Robinhood reviewers highlighted concerns about account access and automated verification processes, Fidelity users frequently mentioned issues related to transaction settlements and account verification delays, and Moomoo users strongly criticized withdrawal procedures and potential scams.

Topic modeling further distilled these negative reviews into specific actionable themes for each app. Fidelity’s users predominantly raised concerns around account usage complexity, fund management issues, verification delays, and transactional errors. Robinhood reviews emphasized automated identity verification frustrations and inadequate personal information handling. Coinbase's negative reviews underscored alarming security issues such as account lockouts, problematic deposit handling, and repeated statement errors. Moomoo's negative feedback mainly highlighted severe issues around withdrawals, allegations of scams, and inadequate customer support.

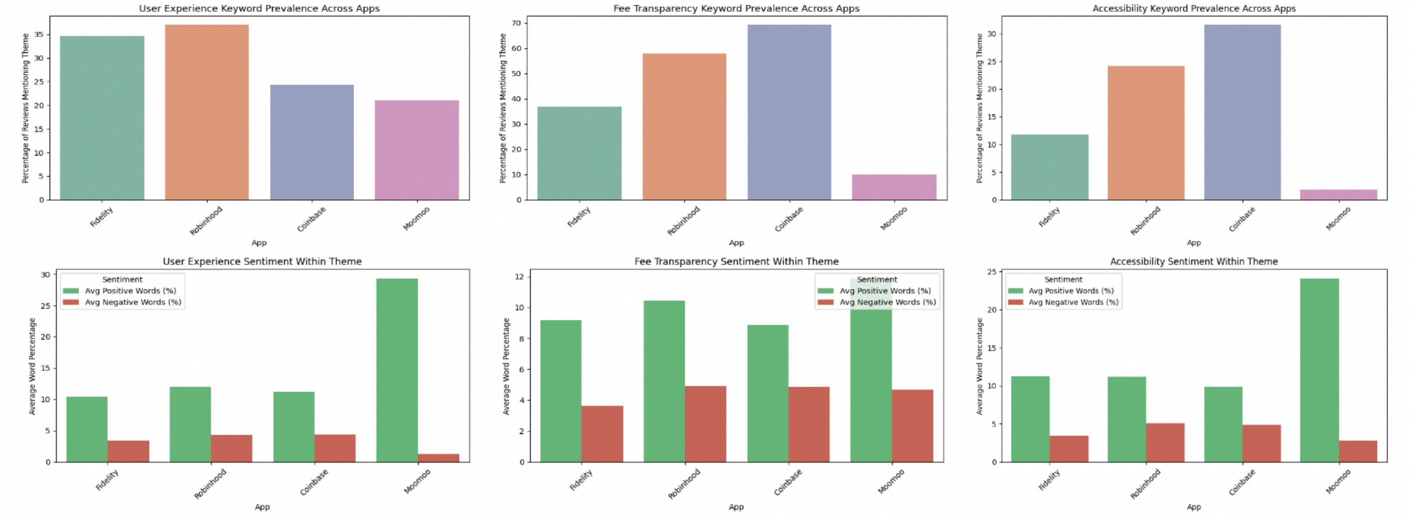


Figure 5: Percentage of Reviews Mentioning Key Themes (Top) and Percentage of Positive and Negative Words (Bottom)

Further examination of user experience, fee transparency, and accessibility through sentiment analysis (Figure 5) provided comparative insights into each app’s perceived performance. Coinbase stood out prominently regarding fee transparency, indicating significant user concerns over hidden fees and unclear financial transactions.

Moomoo, despite overall positive reviews, exhibited critical negative feedback regarding user accessibility and withdrawal functionality, suggesting urgent areas for improvement. Robinhood and Fidelity displayed relatively balanced feedback across these dimensions, with Robinhood facing notable issues related to accessibility and automated processes.

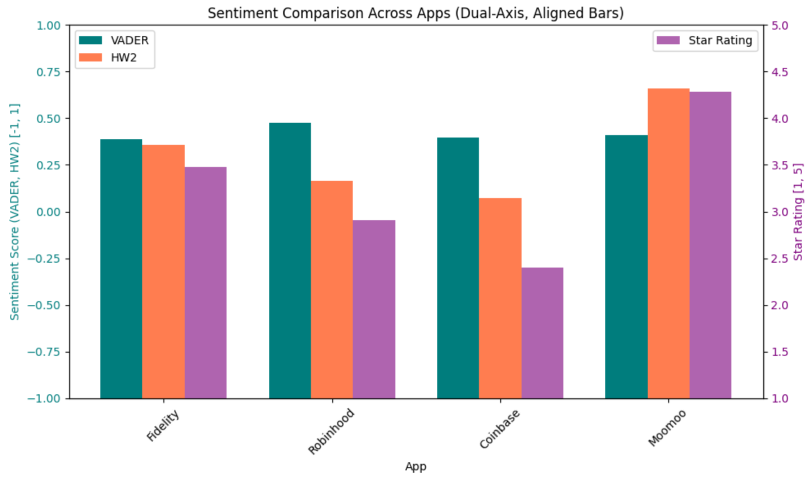


Figure 6: Spread of Star Ratings vs. Sentiment Scores (VADER vs. Hu & Liu (2004))

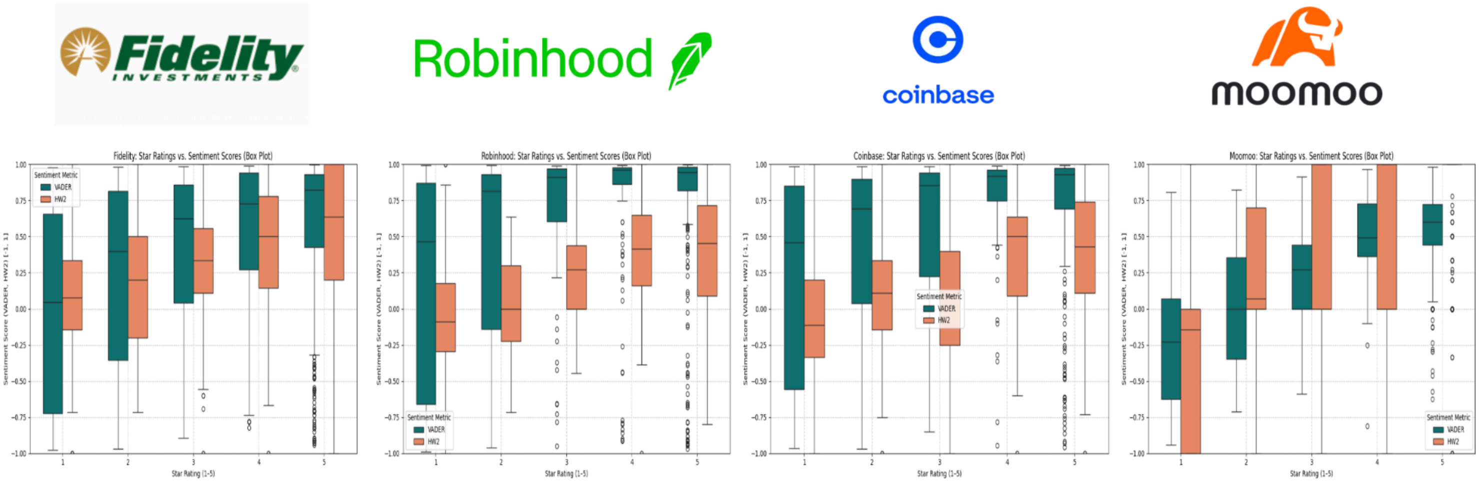
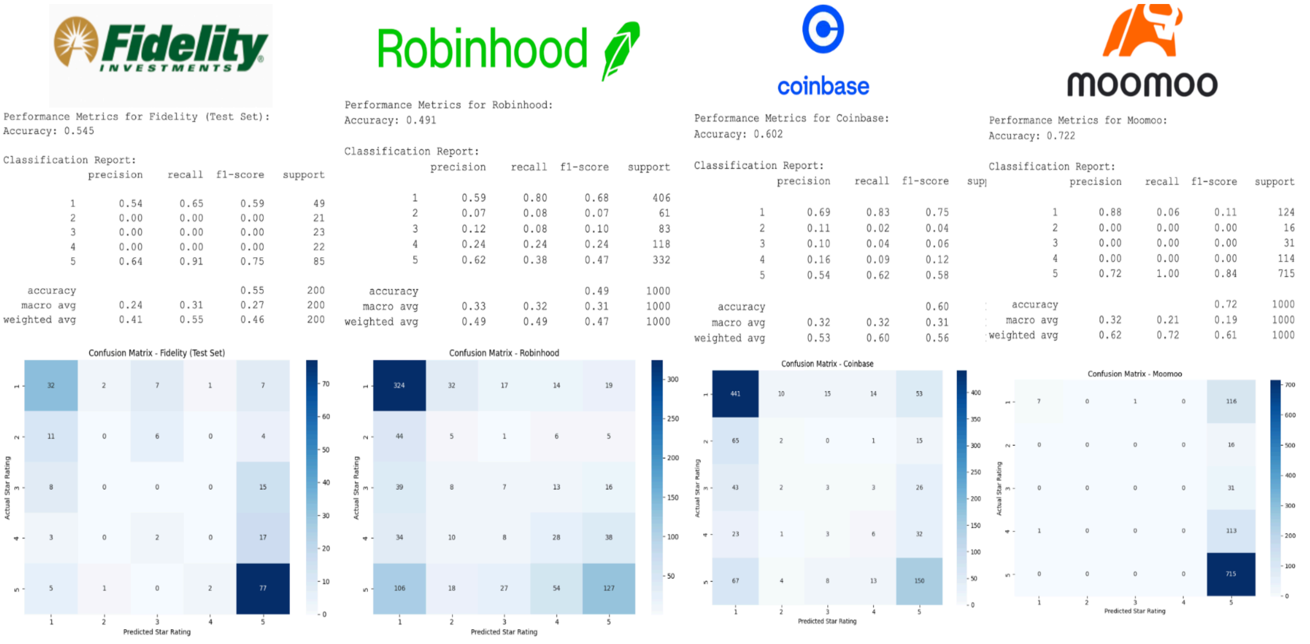


Figure 7: Bar Charts of Star Rating vs. Sentiment Scores (VADER vs. Hu & Liu (2004))

A comparative evaluation of sentiment scores generated from VADER and HW2, based on the simple sentiment calculation initiated by Hu & Liu (2004), against the actual star ratings, shown in Figures 6 and 7 above, underscored a critical insight: textual sentiment, as independently assessed through sophisticated sentiment scoring mechanisms, does not always directly align with numerical star ratings. This finding highlights a substantial challenge in relying exclusively on star ratings for understanding sentiment and isolating positive and negative reviews. The box plots vividly displayed variability and mismatches between sentiment scores and star ratings across all apps, reinforcing the necessity of employing robust sentiment analysis like VADER for more accurate sentiment interpretation.



```
Performance Metrics for Fidelity (Test Set):
Accuracy: 0.545

Classification Report:
              precision    recall  f1-score   support

           1       0.54      0.65      0.59        49
           2       0.00      0.00      0.00        21
           3       0.00      0.00      0.00        23
           4       0.00      0.00      0.00        22
           5       0.64      0.91      0.75        85

    accuracy                           0.55       200
   macro avg       0.24      0.31      0.27       200
weighted avg       0.41      0.55      0.46       200
```

Confusion Matrix - Fidelity (Test Set)

	1	2	3	4	5
1	32	2	7	1	7
2	11	0	6	0	4
3	8	0	0	0	15
4	3	0	2	0	17
5	5	1	0	2	77

```
Performance Metrics for Robinhood:
Accuracy: 0.491

Classification Report:
              precision    recall  f1-score   support

           1       0.59      0.80      0.68       406
           2       0.07      0.08      0.07        61
           3       0.12      0.08      0.10        83
           4       0.24      0.24      0.24       118
           5       0.62      0.38      0.47       332

    accuracy                           0.49      1000
   macro avg       0.33      0.32      0.31      1000
weighted avg       0.49      0.49      0.47      1000
```

Confusion Matrix - Robinhood

	1	2	3	4	5
1	324	32	17	14	19
2	44	5	1	6	5
3	39	8	7	13	16
4	34	10	8	28	38
5	106	18	27	54	127

```
Performance Metrics for Coinbase:
Accuracy: 0.602

Classification Report:
              precision    recall  f1-score   sup

           1       0.69      0.83      0.75
           2       0.11      0.02      0.04
           3       0.10      0.04      0.06
           4       0.16      0.09      0.12
           5       0.54      0.62      0.58

    accuracy                           0.60
   macro avg       0.32      0.32      0.31
weighted avg       0.53      0.60      0.56
```

Confusion Matrix - Coinbase

	1	2	3	4	5
1	441	10	15	14	53
2	65	2	0	1	15
3	43	2	3	3	26
4	23	1	3	6	32
5	67	4	8	13	150

```
Performance Metrics for Moomoo:
Accuracy: 0.722

Classification Report:
              precision    recall  f1-score   support

           1       0.88      0.06      0.11       124
           2       0.00      0.00      0.00        16
           3       0.00      0.00      0.00        31
           4       0.00      0.00      0.00       114
           5       0.72      1.00      0.84       715

    accuracy                           0.72      1000
   macro avg       0.32      0.21      0.19      1000
weighted avg       0.62      0.72      0.61      1000
```

Confusion Matrix - Moomoo

	1	2	3	4	5
1	7	0	1	0	116
2	0	0	0	0	16
3	0	0	0	0	31
4	1	0	0	0	113
5	0	0	0	0	715

Figure 8: Confusion Matrices for Predicted Star Rating

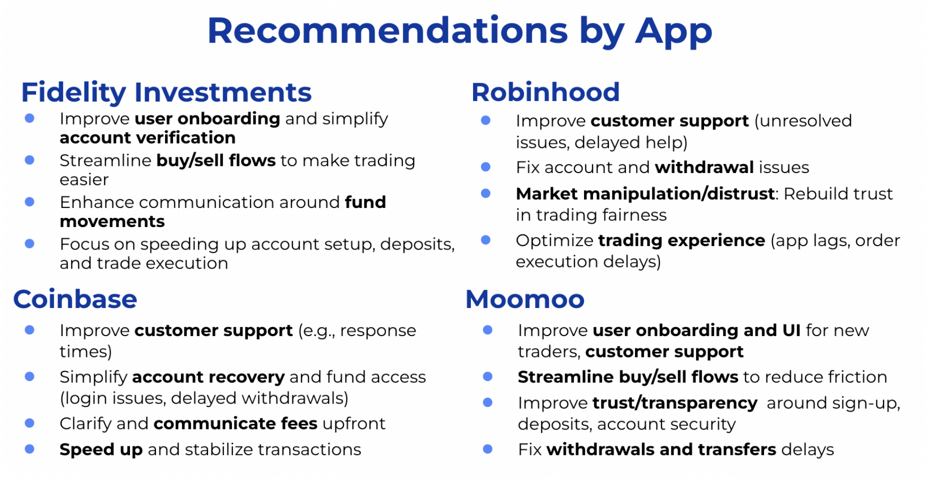
Given this imbalance, interpreting the confusion matrices (Figure 8) from our Random Forest classification requires careful consideration. The classifier was initially trained and evaluated using Fidelity's data, achieving a moderate accuracy of 54.5%. Although the classifier was trained on Fidelity data, we tested its application on the other apps not to generalize performance as each app has different linguistic patterns and user bases, but to further confirm the inconsistencies between written review language and user-assigned star ratings. Performance metrics, including precision, recall, and F1-scores, revealed that the model performed well on extreme ratings (1 and 5 stars) but struggled with intermediary ratings (2, 3, and 4 stars). For instance, Fidelity's confusion matrix indicated robust predictive performance for 5-star ratings (precision 0.64, recall 0.91) and reasonable performance for 1-star ratings (precision 0.54, recall 0.65). However, performance was notably weaker for intermediate ratings due to fewer occurrences and higher variability in review sentiment.

Similar performance trends emerged when applying the Fidelity-trained model to Robinhood, Coinbase, and Moomoo. Robinhood and Coinbase confusion matrices underscored strong predictive capability for the 1-star ratings (precision 0.59 and 0.69, respectively) but poor performance for mid-tier ratings. The predictive model particularly struggled with Coinbase, which had the highest proportion of negative reviews, as indicated by substantial false positives and negatives in the 1-star category. The drop in performance across Robinhood and Coinbase underscores the variability in review text and reinforces that star ratings on their own are not a reliable proxy for user sentiment. Conversely, Moomoo exhibited a notably higher predictive accuracy (72.2%), likely attributable to its skewed positivity; the model effectively recognized the abundant positive sentiment but failed to predict negative or intermediate ratings adequately due to their scarcity. Moomoo's result does not indicate generalizability but further illustrates the challenge of interpreting user sentiment through numerical star ratings alone.

Recommendations by App

The results of our NLP analysis using user reviews for Fidelity, Robinhood, Coinbase, and Moomoo allowed us to derive personalized recommendations for each trading app to address user pain points (Figure 9).

.



Recommendations by App

Fidelity Investments

- Improve **user onboarding** and simplify **account verification**
- Streamline **buy/sell flows** to make trading easier
- Enhance communication around **fund movements**
- Focus on speeding up account setup, deposits, and trade execution

Robinhood

- Improve **customer support** (unresolved issues, delayed help)
- Fix account and **withdrawal** issues
- **Market manipulation/distrust**: Rebuild trust in trading fairness
- Optimize **trading experience** (app lags, order execution delays)

Coinbase

- Improve **customer support** (e.g., response times)
- Simplify **account recovery** and fund access (login issues, delayed withdrawals)
- Clarify and **communicate fees** upfront
- **Speed up** and stabilize transactions

Moomoo

- Improve **user onboarding and UI** for new traders, **customer support**
- **Streamline buy/sell flows** to reduce friction
- Improve **trust/transparency** around sign-up, deposits, account security
- Fix **withdrawals and transfers** delays

Figure 9: Recommendations by App from LDA Most Frequent Topics

For Fidelity, users frequently expressed frustration with account setup and transaction delays. We recommend improving the user onboarding experience, simplifying account verification, and streamlining buy/sell flows. Enhanced communication around fund movements and faster execution of deposits and trades would also improve user satisfaction.

Robinhood users often cited unresolved customer service issues and concerns around fairness in trading. Thus, improving support quality, resolving withdrawal delays, addressing market manipulation concerns, and optimizing the trading experience (e.g., reducing app lag and execution delays) are key areas for improvement.

Coinbase users frequently experienced issues with customer support (e.g., response times) and transaction reliability. To address these, we suggest streamlining support response, targeting bugs pertaining to login issues, and expediting withdrawals and transactions. Additionally, many users were unclear about the expected fees in the trading process. Thus, clearly communicating fees upfront will significantly increase user trust.

Finally, since it was just released in 2022 and has had fewer updates, Moomoo would benefit from onboarding and UI enhancements for new users. Like the previous three apps, improved customer support, increased transparent communication around trust and security, and faster processing of withdrawals and transfers are also key areas to work on. These recommendations directly address the most frequent and impactful user concerns found in our NLP analysis.

Strengths, Limitations, and Future Directions

Building a project around analyzing user reviews to make development and business decisions has multiple pros and cons. User-generated reviews are sources of rich, real-world data that are often passionate and emotionally charged, which indicates a perfect source for sentiment and topic extractions; there is also a plethora of easily accessible reviews across most App Store apps using scraper packages, which allows for replicability of this project to analyze any app. There is a broader variety of available literature on how to think about the psychology behind user reviews and how to analyze them, making this an interesting project to continue the discussion. At the same time, user-generated text is often messy and contains many difficult things to analyze, such as sarcasm, slang, multilingual switching, misspellings, or grammatical issues. As discussed in our project, star ratings do not always clearly match the sentiment of review text. Lastly, reviews can be biased, either skewing unfavorable due to negativity bias, with bots leaving five or one-star reviews, or users being incentivized to leave higher reviews.

While our analysis provided meaningful, actionable insights, several other limitations should be addressed. First, there is great subjectivity in interpreting topics from LDA modeling, which may affect consistency across reviewers. Although we conducted manual spot checks to validate topic coherence, different reviewers might assign slightly different interpretations to the same cluster of keywords. Future work could incorporate more systematic approaches, such as coherence scores, to reduce this subjectivity. Our data is also highly likely to be influenced by self-selection bias, as users who leave reviews are likelier to have had negative experiences. As a result, some user experiences may be underrepresented or entirely absent from the dataset. Lastly, Moomoo had fewer reviews due to its relatively recent release, limiting comparative analysis due to the imbalanced review volume across apps. This may limit the generalizability of our findings for Moomoo.

This NLP analysis of trading app reviews lays the groundwork for promising future work. To investigate user concerns and sentiment over time, such as before and after app releases, and to identify effects of feature updates or emerging trends, we suggest tracking these temporal changes using time series analysis. BERT-based models could also be integrated to enhance sentiment classification by capturing more nuanced language and context. For example, RoBERTa is strong in language understanding and nuanced sentiment, and FinBERT is a pre-trained, domain-specific NLP model for financial sentiment, making it great for trading and investing language. Finally, a more granular sentiment analysis tied to specific app features (e.g., trading tools, verification processes, or customer service) could yield more detailed insights for product improvement.

Conclusion

Our project demonstrates how NLP can be utilized to transform unstructured App Store review data into meaningful insights for product improvement. By first attempting to predict star ratings using a supervised Random Forest classifier, we discovered a misalignment between numerical ratings and written sentiments, highlighting the limitations of star ratings as a proxy for sentiment and demonstrating the need for a more sophisticated analysis. The Random Forest classifier was used as a diagnostic to test for latent alignment rather than building a functional rating predictor. This gap was addressed by applying VADER sentiment analysis to more consistently evaluate the true tone in each review, improving our Random Forest classifier diagnostic performance. Focusing on reviews that were revealed to be negative, we then used LDA to uncover prominent user themes and pain points. These include issues such as login failures, poor customer support, and difficult-to-navigate user interfaces – all insights that can create actionable recommendations for developers. Issues prominently expressed in reviews go beyond just UI and UX and cover many critical app pain points. This project illustrates the value of combining unsupervised and supervised natural language processing techniques to extract strategic, data-driven guidance from noisy user-generated content.

Works Cited

Chaudhry, S., & Chinmay, K. (2021, June 6). *Design Patterns of Investing Apps and Their Effects on Investing Behaviors*. Designing Interactive Systems Conference 2021. https://doi.org/10.1145/3461778.3462008.

Egger, R., & Yu, J. (2022, May 6). *A topic modeling comparison between LDA, NMF, Top2Vec, and Bertopic to demystify twitter posts*. Frontiers in sociology. https://pmc.ncbi.nlm.nih.gov/articles/PMC9120935/

Forbord, K. (2022, May 6). *Why are users continuously abandoning onboarding to financial...* Signicat. https://www.signicat.com/the-battle-to-onboard-2022/abandonment-to-financial-service-onboarding-over-the-years

Guzman, E., & Maalej, W. (2014, August 25). *How do users like this feature? A fine grained sentiment analysis of App Reviews*. IEEE Explore. https://ieeexplore.ieee.org/document/6912257/

Hu, M., & Liu, B. (2004, August 22). *Mining and Summarizing Customer Reviews.* University of Illinois at Chicago, Department of Computer Science. https://www.cs.uic.edu/~liub/publications/kdd04-revSummary.pdf

Isnan, M., Elwirehardja, G. N., & Pardamean, B. (2023, November 25). *Sentiment Analysis for TikTok review using Vader sentiment and SVM model*. Procedia Computer Science. https://www.sciencedirect.com/science/article/pii/S1877050923016800

Kumar Gupta, R. (2022, April 2). *Prediction of research trends using LDA based topic modeling*. Science Direct. https://www.sciencedirect.com/science/article/pii/S2666285X22000206

Lush, M., Fontes, A., Zhu, M., Valdes, O., & Mottola, G. (2021, February 1). *Investing 2020: New accounts and the people who ...* Finra Investor Education Foundation. https://www.finrafoundation.org/sites/finrafoundation/files/investing-2020-new-accounts-and-the-people-who-opened-them_1_0.pdf

Shah, F. A., Sabir, A., & Sharma, R. (2024, September 2). *A Fine-grained Sentiment Analysis of App Reviews using Large Language Models: An Evaluation Study*. ResearchGate. https://www.researchgate.net/publication/282272480_How_Do_Users_Like_This_Feature_A_Fine_Grained_Sentiment_Analysis_of_App_Reviews

Xu, Y., Liu, Y., Xu, H., & Tan, H. (2024, July 1). *AI-driven UX/UI Design: Empirical Research and Applications in FinTech*. ResearchGate. https://www.researchgate.net/publication/382956602_AI-Driven_UXUI_Design_Empirical_Research_and_Applications_in_FinTech