Mining Shopee User Reviews: Sentiment Analysis and Topic Modeling Perspective

YOUHUAN_24100710

ASSIGNMENT

LECTURER: TS. DR. NOR HASLIZA BINTI MD SAAD

ACADEMIC SESSION 2024/2025

MASTER IN BUSINESS ANALYTICS ABM503 – WEB & SOCIAL MEDIA ANALYTICS

Submission Date : 24 June 2025

Abstract

With the rapid expansion of mobile commerce in Southeast Asia, Shopee has emerged as one of the leading platforms, attracting millions of app-based reviews on Google Play. These user-generated reviews contain valuable feedback about product experience, service quality, app performance, and overall customer satisfaction. This study aims to explore and analyze English reviews of the Shopee app using a hybrid approach that combines sentiment analysis and topic modeling. Specifically, we employ the VADER tool to classify user sentiment into positive, neutral, and negative categories, and implement Latent Dirichlet Allocation (LDA) to uncover eight underlying topics within the reviews.

A dataset of 10,000 Shopee reviews was collected and cleaned using a standardized preprocessing pipeline. Results show that while 72% of the reviews express positive sentiment, a significant portion of user dissatisfaction is concentrated around themes such as delivery delays, customer service issues, and app malfunctions. The VADER sentiment classifier aligns closely with rating-based sentiment in most cases but shows greater sensitivity to textual tone. The integration of LDA and VADER further allows us to identify emotion-rich topics, providing insights into user pain points and satisfaction drivers.

This research contributes both theoretically and practically. Theoretically, it demonstrates the effectiveness of combining lexicon-based sentiment analysis with unsupervised topic modeling in mobile app review analysis. Practically, it offers Shopee actionable insights to enhance user experience, prioritize service improvements, and build sentiment-aware customer management systems. Future research may incorporate multilingual reviews, advanced deep learning models, or cross-platform comparisons.

CONTENT

Chapter 1: Introduction

1.1 Research Background

With the rapid development of mobile internet and smartphone usage, more consumers are turning to mobile applications (apps) for shopping, payment, and logistics. The user experience on e-commerce platforms' mobile interfaces has thus become a key factor in influencing user satisfaction and loyalty.

User reviews submitted via app stores, such as Google Play, are not merely expressions of opinion—they constitute User-Generated Content (UGC) that significantly impacts the purchase decisions of potential buyers. These reviews contain rich feedback on aspects such as product features, service quality, and usability, offering valuable insights for both platform operators and academic researchers.

Shopee, as one of Southeast Asia's most popular e-commerce platforms, generates a high volume of mobile-based user reviews. Analyzing this content holds practical significance for understanding user sentiment and behavior. In particular, the ability to efficiently extract key concerns, pain points, and emotional fluctuations from large-scale natural language feedback has become increasingly vital for business operations and research in digital marketing and customer experience management.

1.2 Research Objectives and Questions

This study focuses on analyzing English reviews of the Shopee app on Google Play using natural language processing (NLP) techniques. It integrates sentiment analysis and topic modeling to uncover core themes and emotional trends, with a further aim to enhance interpretability through data visualization.

The main objectives of this research are:

- To classify Shopee reviews into positive, neutral, and negative sentiments using the VADER sentiment analysis tool;
- To identify latent topics from review texts using Latent Dirichlet Allocation (LDA);
- To explore the relationship between emotional tendencies and specific review topics;

- To visualize sentiment trends and key topics to support practical business insights.

Key research questions include:

- What are the dominant topics in Shopee user reviews?
- How does sentiment vary across different themes?
- Which topics are more likely to trigger negative feedback?
- Are there observable trends in user sentiment over time?

1.3 Research Significance

Theoretical Contribution

This study combines lexicon-based sentiment analysis (VADER) with probabilistic topic modeling (LDA), offering a hybrid approach to review mining. It contributes to interdisciplinary research across text mining, social media analytics, and consumer behavior, especially in the context of unstructured review data.

Practical Value

From a business perspective, the findings provide actionable methods for extracting insights from large volumes of customer feedback. Platform operators can use the results to improve service strategies, monitor app performance, and identify operational weaknesses. The sentiment and topic trends also assist developers and marketers in optimizing product features and communication strategies.

1.4 Research Workflow Overview

The research adopts a standardized text mining pipeline, as outlined below:

1. **Data Collection**: User reviews are scraped from the Shopee app's Google Play page using Selenium.
2. **Data Cleaning**: Includes deduplication, removal of null values, normalization, tokenization, stopword removal, and lemmatization.

3. **Sentiment Analysis**: Each review is analyzed using VADER; based on the compound score, labels are assigned as positive, neutral, or negative.
4. **Topic Modeling**: Cleaned reviews are used to train an LDA model, generating 8 distinct topics.
5. **Visualization**: Results are presented via sentiment distribution bar charts, temporal trend line plots, word clouds, and an interactive pyLDAvis interface.

1.5 Structure of the Thesis

This thesis is structured as follows:

- **Chapter 1 – Introduction**: Outlines the research background, objectives, significance, and design;
- **Chapter 2 – Literature Review**: Reviews existing studies on sentiment analysis, topic modeling, and user behavior in e-commerce;
- **Chapter 3 – Methodology**: Details the data sources, analytical framework, modeling tools, and visualization techniques;
- **Chapter 4 – Findings**: Presents key results, including sentiment distributions, LDA topic keywords, and time-based trends;
- **Chapter 5 – Discussion**: Interprets findings and proposes business implications and strategic recommendations;
- **Chapter 6 – Conclusion**: Summarizes the study, highlights limitations, and suggests directions for future research.

Chapter 2: Literature Review

This chapter reviews the theoretical foundations and relevant research related to this study, including sentiment analysis approaches, the principles of topic modeling using LDA, the behavioral impact of user-generated reviews, and key text processing techniques. These reviews serve to contextualize the current research and guide its methodological framework.

2.1 Overview of Sentiment Analysis

2.1.1 DEFINITION AND CLASSIFICATION

Sentiment analysis, a central task in Natural Language Processing (NLP), aims to identify and extract subjective information from text, including sentiment polarity (positive, neutral, or negative) and emotional intensity. Based on methodology, sentiment analysis can be broadly categorized into two types: lexicon-based and machine learning-based approaches.

2.1.2 LEXICON-BASED METHODS: VADER

Lexicon-based sentiment analysis relies on predefined dictionaries that associate words with sentiment scores. VADER (Valence Aware Dictionary and sEntiment Reasoner) is a widely adopted lexicon-based tool optimized for analyzing informal, social-media-style text such as tweets and online reviews.

Advantages of VADER include:

- Optimized for short, noisy text typical in reviews or app feedback;
- Accounts for syntactic and semantic nuances such as capitalization, punctuation, intensifiers, and negation;
- Outputs a compound score as an overall sentiment metric.

Due to its lightweight design and lack of training data requirements, VADER is ideal for small-scale projects or scenarios with limited resources. However, it struggles to handle nuanced context such as sarcasm or idiomatic expressions.

2.1.3 MACHINE LEARNING-BASED METHODS (BRIEF OVERVIEW)

Machine learning models treat sentiment classification as a supervised learning problem. Common algorithms include Logistic Regression (LR), Naive Bayes (NB), Support Vector Machines (SVM), and Random Forests (RF). These models generally offer higher accuracy but require large labeled datasets for training. In contrast, VADER offers simplicity and ease of deployment.

2.2 Topic Modeling with LDA

Topic modeling is an unsupervised method used to uncover hidden thematic structures in large text corpora. Latent Dirichlet Allocation (LDA), proposed by Blei et al. (2003), remains the most well-known algorithm in this domain.

LDA assumes that each document (e.g., a user review) is a mixture of latent topics, and each topic is characterized by a distribution over keywords. This modeling allows researchers to:

- Automatically extract dominant discussion themes in reviews;
- Reduce high-dimensional text data into interpretable topics;
- Detect key user concerns such as pricing, logistics, service quality, and usability.

LDA has been widely applied in e-commerce review mining for purposes such as:

- Extracting themes linked to product satisfaction;
- Identifying negative feedback drivers;
- Mapping emotion-topic intersections.

2.3 User Reviews and Consumer Behavior

User reviews are a form of electronic word-of-mouth (eWOM) and significantly influence consumer decisions. Research indicates that:

- Sentiment polarity affects consumer trust and purchase intention;
- Volume, emotional intensity, and diversity of reviews impact perceived product quality;
- Negative reviews generally exert a stronger impact due to negativity bias.

Moreover, reviews often reveal users' emotional states, expectations, and pain points, making them valuable for improving customer understanding and platform strategy.

2.4 Text Processing and Modeling Techniques

2.4.1 TEXT PREPROCESSING

Before performing sentiment analysis or topic modeling, user reviews must undergo preprocessing. Common steps include:

- Lowercasing;
- Removing punctuation and special symbols;
- Tokenization;
- Removing stopwords (e.g., “the”, “is”);
- Lemmatization to reduce words to their base forms.

Due to the informal nature of online reviews—with frequent slang, typos, emojis, and abbreviations—flexible preprocessing is essential.

2.4.2 FEATURE EXTRACTION: TF-IDF AND BOW

Converting text into numerical features is a critical step in text mining:

- **Bag-of-Words (BoW)**: Converts text into word frequency vectors without considering word order;
- **TF-IDF (Term Frequency-Inverse Document Frequency)**: Measures a word’s importance relative to a document and corpus, helping emphasize informative terms.

In this study, TF-IDF is used as input for machine learning sentiment classification tasks.

2.4.3 LDA FOR TEXT DIMENSIONALITY REDUCTION

LDA serves as a probabilistic dimensionality reduction technique, offering interpretable topic structures compared to traditional clustering. In review analysis, LDA enables:

- Clustering reviews into thematic groups;
- Attaching topic tags to reviews for further analysis such as sentiment-topic correlation and frequency statistics.

2.5 Summary

This chapter reviewed foundational theories and empirical studies on sentiment analysis, topic modeling, and consumer behavior in the context of user-generated reviews. By adopting VADER and LDA, this study aims to both quantify sentiment polarity and uncover hidden topics in Shopee reviews. The combined use of these tools offers a structured and insightful framework for businesses to interpret customer feedback and make data-driven decisions.

Chapter 3: Methodology

This chapter outlines the methodological framework adopted for analyzing user reviews of the Shopee app on the Google Play Store. The research process includes data collection, data cleaning, text preprocessing, sentiment analysis, topic modeling, and result visualization. All analyses were conducted using Python, incorporating various natural language processing and visualization libraries to ensure analytical rigor and interpretability.

3.1 Data Source and Description

The dataset used in this study consists of English-language user reviews of the Shopee app collected from the Google Play Store. An automated scraping tool was employed to extract the data, resulting in a total of 10,000 valid reviews. Each review entry contains the following fields:

- **reviewId**: Unique identifier for the review
- **userName**: Username of the reviewer
- **Review Text**: Main content of the review
- **score**: Star rating provided by the user (ranging from 1 to 5)
- **Review Date**: Date when the review was posted

reviewId	userName	reviewText	score	thumbsU	reviewCrea	reviewDate
939b6fe0-bab	samantha loh	shopee platform is more diverse, not crowded with mainly 90% china products.	5	0	3.51.33	2025/6/17 15:03
0be9046f-189	M M	Great app but they need to fix putting all notifications on one category, which includ	2	0	3.51.33	2025/6/17 11:57
ea572dc6-5fc	subra maniam	fantastic	5	0	3.51.33	2025/6/17 9:45
ed1e2243-d08	Zahid Hossain	good	5	0	3.51.33	2025/6/16 23:07
47480e0c-dd5	Shahriar Najmul	good apps	5	0	3.51.33	2025/6/16 22:54
9ea7a34b-5b3	Mohamed justice	Great platform with good service rendered.	5	0	3.51.33	2025/6/16 21:54
433172a4-a57	Jo Khoo	User friendly app	5	0	3.51.33	2025/6/16 20:12
cd21c7fb-5a0	ishak	Very good. Everything is good.	5	0	3.51.33	2025/6/16 13:56
155a44bb-658	Mamin Marvel	people need to learn to use shopee correctly.. smartly buy the good products.. theres	5	0	3.51.33	2025/6/16 11:45
eb7ee255-82b	Saiful Islam	very good apps	5	0		2025/6/15 22:50

3.2 Research Workflow Design

The overall research workflow of this study is illustrated as follows:

1. **Data Collection**: Reviews were scraped from the Shopee app's Google Play Store page using Selenium to simulate browser behavior.
2. **Data Preprocessing**: Missing values were removed, review text was standardized, and date formats were cleaned and unified.
3. **Sentiment Analysis**: Each review was classified into a sentiment category (positive, neutral, negative) using the VADER tool.
4. **Topic Modeling**: An unsupervised Latent Dirichlet Allocation (LDA) algorithm was applied to extract latent themes from the review corpus.
5. **Visualization**: A series of plots were generated, including sentiment distribution charts, monthly trend lines, word clouds, and interactive topic maps.

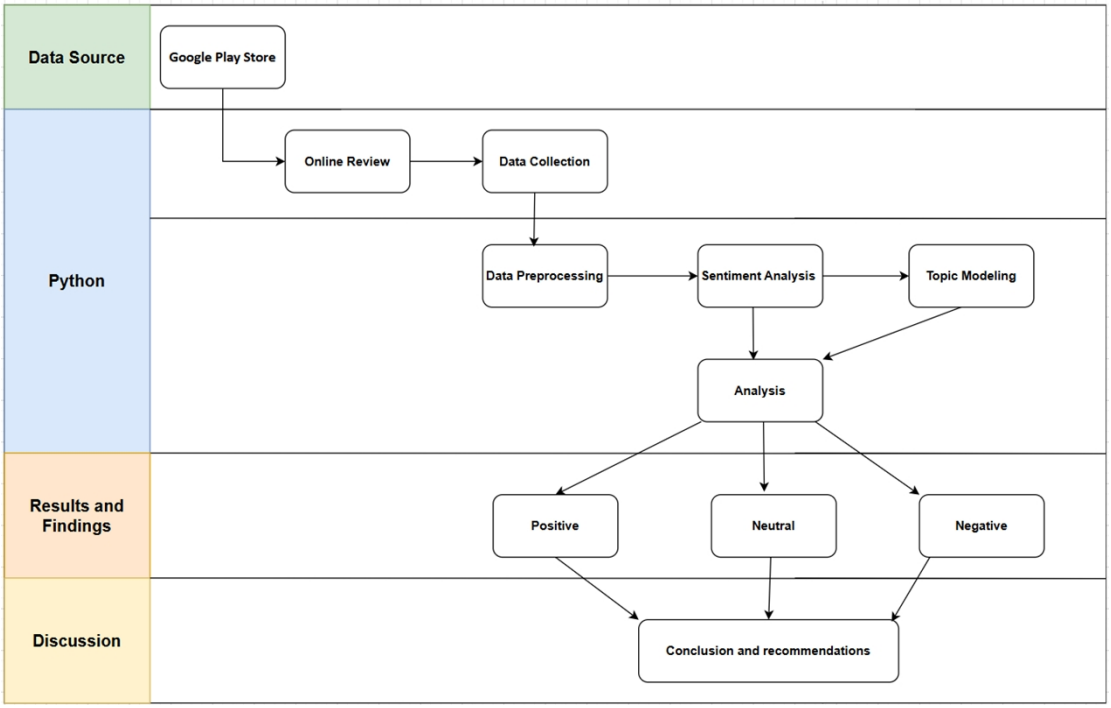


Figure 3.1 Flow Chart

3.3 Data Preprocessing

3.3.1 CLEANING AND FILTERING

The raw review dataset underwent the following cleaning steps:

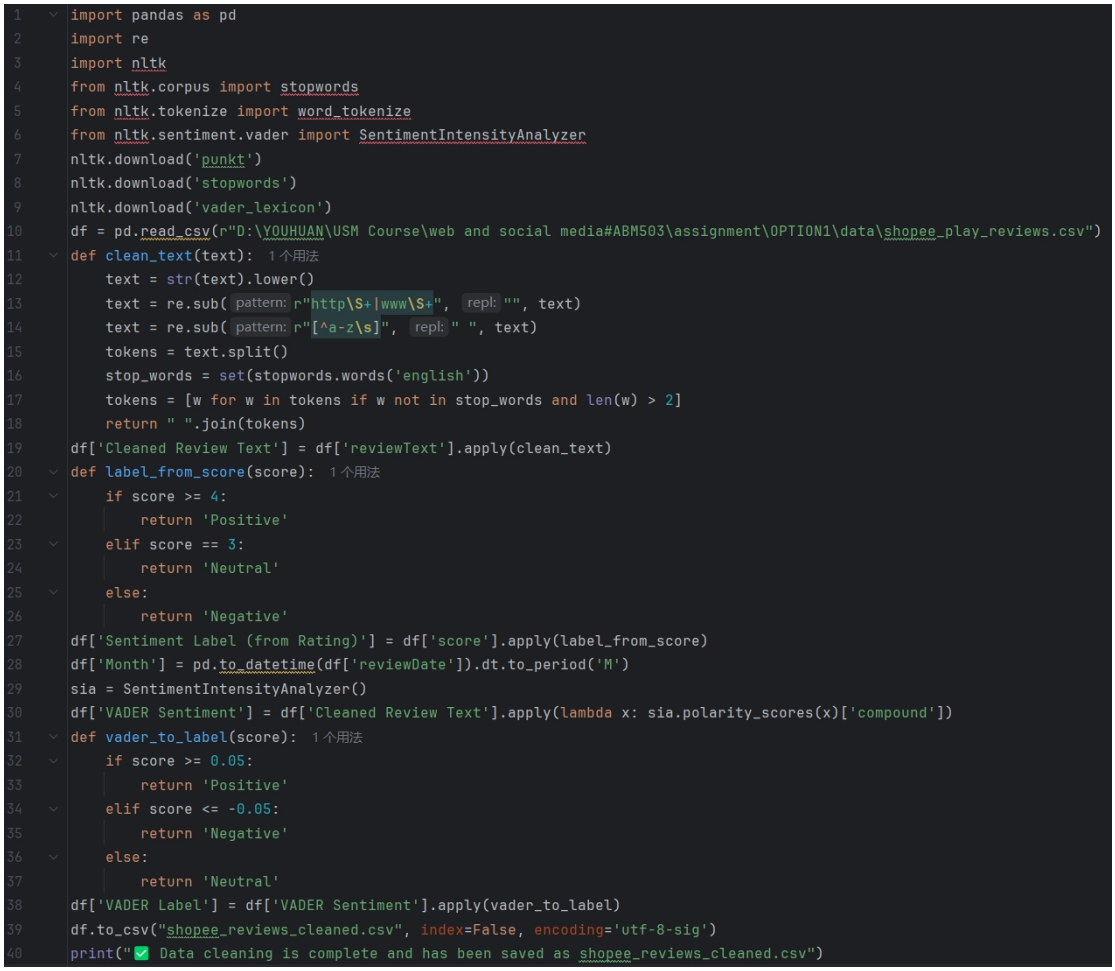
- Removal of empty entries and records without ratings;
- Filtering to retain only English-language reviews;
- Conversion of the **Review Date** field into standard date format, with an additional **Month** field extracted for trend analysis.

3.3.2 TEXT NORMALIZATION

To prepare the textual data for sentiment analysis and topic modeling, a custom preprocessing function (clean_text) was applied to the **Review Text** field. The following operations were performed:

- Lowercasing of all text;
- Removal of URLs, HTML tags, punctuation, and numerical characters;
- Tokenization using word_tokenize;
- Elimination of stopwords using standard English stopword lists;
- Lemmatization to reduce words to their base forms.

The cleaned output was saved in a new column titled **Cleaned Review Text**, which served as the primary input for subsequent text analysis tasks.



```python
import pandas as pd
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.sentiment.vader import SentimentIntensityAnalyzer
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('vader_lexicon')
df = pd.read_csv(r"D:\YOUHUAN\USM Course\web and social media#ABM503\assignment\OPTION1\data\shopee_play_reviews.csv")
def clean_text(text):  1 个用法
    text = str(text).lower()
    text = re.sub( pattern: r"http\S+|www\S+",  repl: "", text)
    text = re.sub( pattern: r"[^a-z\s]",  repl: " ", text)
    tokens = text.split()
    stop_words = set(stopwords.words('english'))
    tokens = [w for w in tokens if w not in stop_words and len(w) > 2]
    return " ".join(tokens)
df['Cleaned Review Text'] = df['reviewText'].apply(clean_text)
def label_from_score(score):  1 个用法
    if score >= 4:
        return 'Positive'
    elif score == 3:
        return 'Neutral'
    else:
        return 'Negative'
df['Sentiment Label (from Rating)'] = df['score'].apply(label_from_score)
df['Month'] = pd.to_datetime(df['reviewDate']).dt.to_period('M')
sia = SentimentIntensityAnalyzer()
df['VADER Sentiment'] = df['Cleaned Review Text'].apply(lambda x: sia.polarity_scores(x)['compound'])
def vader_to_label(score):  1 个用法
    if score >= 0.05:
        return 'Positive'
    elif score <= -0.05:
        return 'Negative'
    else:
        return 'Neutral'
df['VADER Label'] = df['VADER Sentiment'].apply(vader_to_label)
df.to_csv("shopee_reviews_cleaned.csv", index=False, encoding='utf-8-sig')
print("✅ Data cleaning is complete and has been saved as shopee_reviews_cleaned.csv")
```

Figure 3.2 Data Cleaning

Row data	Cleaned data
shopee platform is more diverse, not crowded with mainly 90% china products.	shopee platform diverse crowded mainly china products
Very good. Everything is good.	good everything good
very good apps	good apps
very good , able to find most of the things I am looking for , great job .	good able find things looking great job
Love the self collection option. So convenient and useful	love self collection option convenient useful

Figure 3.3 Row data &Cleaned Data Compared

3.4 Sentiment Analysis Method (VADER)

This study employed the **VADER** (Valence Aware Dictionary and sEntiment Reasoner) sentiment analysis tool from the NLTK library to assess the emotional polarity of each user review. For every **Review Text**, VADER generates four sentiment metrics:

- **neg**: Negative sentiment intensity

- **neu**: Neutral sentiment intensity
- **pos**: Positive sentiment intensity
- **compound**: A normalized compound score ranging from −1 (most negative) to +1 (most positive)

Based on the compound score, sentiment labels were assigned using the following thresholds:

- **Positive**: compound ≥ 0.05
- **Neutral**: −0.05 < compound < 0.05
- **Negative**: compound ≤ −0.05

The resulting labels were stored in a new column, **VADER Label**, and were used for further sentiment distribution and trend analysis.

Compound Range	**Emotion**
≥ 0.05	Positive
≤ -0.05	Negative
Others	Neutral

Figure 3.4 Compound Range & Emotion

ReviewText	VADER Sentiment	VADER Label
fantastic	0.5574	Positive
good	0.4404	Positive
good apps	0.4404	Positive
Great platform with good service rendered.	0.7906	Positive
User friendly app	0.4939	Positive
Very good. Everything is good.	0.7003	Positive
very good apps	0.4404	Positive
good service	0.4404	Positive
good	0.4404	Positive
great n fast delivery	0.6249	Positive

Very lag now	-0.34	Negative
fast delivery	0	Neutral

Figure 3.5 Vader Sentiment & Vader Label

3.5 LDA Topic Modeling

To uncover the major topics embedded within user reviews, this study implemented **Latent Dirichlet Allocation (LDA)** using the **Gensim** library. The modeling process consisted of the following key steps:

1. **Corpus and Dictionary Construction**:

   The tokenized and preprocessed texts (from the Cleaned Review Text column) were used to create a dictionary using corpora.Dictionary(), and then transformed into a Bag-of-Words corpus using doc2bow().

2. **Model Training**:

   An LDA model was trained on the corpus with the number of topics set to **8**, using default hyperparameters (passes=10, random_state=42) to ensure stability and reproducibility.

3. **Topic Extraction**:

   For each topic, the top 10 representative keywords were extracted, providing interpretable semantic cues for identifying the dominant themes in the corpus.

The extracted topics reflected user concerns across a range of areas including delivery experience, app usability, customer service, pricing, and shopping satisfaction.

```python
import pandas as pd
import gensim
from gensim import corpora
from wordcloud import WordCloud
import matplotlib.pyplot as plt
import pyLDAvis.gensim_models

df = pd.read_csv("shopee_reviews_cleaned.csv")

texts = df['Cleaned Review Text'].dropna().apply(lambda x: x.split()).tolist()
dictionary = corpora.Dictionary(texts)
corpus = [dictionary.doc2bow(text) for text in texts]

lda_model = gensim.models.ldamodel.LdaModel(corpus=corpus,
                                            id2word=dictionary,
                                            num_topics=8,
                                            random_state=42,
                                            passes=10,
                                            per_word_topics=True)

print("\n📌 Top Words per Topic:")
for i, topic in lda_model.show_topics(formatted=False):
    words = [word for word, _ in topic]
    print(f"🟣 Topic {i + 1}: {', '.join(words)}")

for i, topic in lda_model.show_topics(formatted=False):
    word_freq = {word: weight for word, weight in topic}
    wc = WordCloud(width=800, height=400, background_color='white').generate_from_frequencies(word_freq)
    plt.figure(figsize=(8, 4))
    plt.imshow(wc, interpolation='bilinear')
    plt.axis("off")
    plt.title(f"Topic {i + 1} WordCloud")
    plt.tight_layout()
    plt.savefig(f"topic_{i + 1}_wordcloud.png")
    plt.close()
print("✅ All the word clouds have been created!")
print("📊 Generating the interaction diagram... Please wait...")
import os
import tempfile
os.environ['JOBLIB_TEMP_FOLDER'] = tempfile.mkdtemp(prefix="lda_temp_", dir="C:/Temp")
vis = pyLDAvis.gensim_models.prepare(lda_model, corpus, dictionary)
pyLDAvis.save_html(vis, "shopee_lda_gensim_visualization.html")
print("✅ The visual graph has been saved as shopee_lda_gensim_visualization.html")
```

Figure 3.6 LDA Topic Modeling

3.5.1 CORPUS AND MODEL INPUT CONSTRUCTION

The preprocessed review data were first imported using **pandas**, and tokenized by applying the .split() function to the Cleaned Review Text column. This generated a list of tokenized documents (texts). Subsequently, a **Gensim** dictionary was created via corpora.Dictionary(texts), which mapped each unique token to an integer ID. The tokenized texts were then converted into a **Bag-of-Words (BoW)** format using the doc2bow() method, resulting in a sparse representation suitable for input into the LDA model.

This process established the foundational inputs—**dictionary** and **corpus**—for probabilistic topic modeling.

```
texts = df['Cleaned Review Text'].dropna().apply(lambda x: x.split()).tolist()
dictionary = corpora.Dictionary(texts)
corpus = [dictionary.doc2bow(text) for text in texts]
```

Figure 3.7 Corpus And Model Input Construction

3.5.2 MODEL TRAINING AND TOPIC EXTRACTION

After preparing the corpus and dictionary, the **LDA model** was trained using Gensim's LdaModel function. The number of topics was set to **8**, with **10 passes** over the corpus to ensure stable convergence. A fixed random_state was applied to enhance reproducibility.

Upon completion of training, the model generated a set of **top 10 keywords** for each topic, reflecting the dominant terms within each latent theme. These extracted keywords serve as a basis for subsequent visualization and thematic interpretation of the user reviews.

```
lda_model = gensim.models.ldamodel.LdaModel(corpus=corpus,
                                            id2word=dictionary,
                                            num_topics=8,
                                            random_state=42,
                                            passes=10,
                                            per_word_topics=True)

print("\n📌 Top Words per Topic:")
for i, topic in lda_model.show_topics(formatted=False):
    words = [word for word, _ in topic]
    print(f"🟣 Topic {i + 1}: {', '.join(words)}")
```

Figure 3.8 Model Training And Topic Extraction

3.5.3 VISUALIZATION IMPLEMENTATION

To enhance the interpretability of the extracted topics, this study employs both WordCloud and pyLDAvis for static and interactive visualizations. WordClouds illustrate the most frequent terms for each topic in a visually intuitive format, while pyLDAvis generates an interactive HTML interface that allows users to explore the distribution and overlap of topics and keywords within the corpus.

WordCloud Static Word Cloud Chart:

```python
for i, topic in lda_model.show_topics(formatted=False):
    word_freq = {word: weight for word, weight in topic}
    wc = WordCloud(width=800, height=400, background_color='white').generate_from_frequencies(word_freq)
    plt.figure(figsize=(8, 4))
    plt.imshow(wc, interpolation='bilinear')
    plt.axis("off")
    plt.title(f"Topic {i + 1} WordCloud")
    plt.tight_layout()
    plt.savefig(f"topic_{i + 1}_wordcloud.png")
    plt.close()
```

Figure 3.9 Visualization Implementation

PyLDAvis interactive visualization diagram:

```python
print("✅ All the word clouds have been created!")
print("📊 Generating the interaction diagram... Please wait...")
import os
import tempfile
os.environ['JOBLIB_TEMP_FOLDER'] = tempfile.mkdtemp(prefix="lda_temp_", dir="C:/Temp")
vis = pyLDAvis.gensim_models.prepare(lda_model, corpus, dictionary)
pyLDAvis.save_html(vis, "shopee_lda_gensim_visualization.html")
print("✅ The visual graph has been saved as shopee_lda_gensim_visualization.html")
```

Figure 3.10 PyLDAvis Interactive Visualization Diagram

A total of 8 topics were successfully generated, along with corresponding keyword lists, static word clouds (topic_1_wordcloud.png to topic_8_wordcloud.png), and an interactive visualization page (shopee_lda_gensim_visualization.html). These outputs provide intuitive and insightful representations of topic patterns within the user reviews.

Number	Topic Name	Top 10 Keywords
Topic 1	Problems with the APP usage	app, shopee, back, account, review, phone, still, use, even, get
Topic 2	After-sales and logistics services	refund, shopee, delivery, service, item, customer, time, seller, return, order
Topic 3	Product content and price evaluation	products, nice, product, sellers, price, shopee, like, variety, items, prices
Topic 4	Cost-effectiveness and user satisfaction	good, easy, delivery, fast, use, service, far, happy, shopee, app
Topic 5	Payment and Account Experience	app, shopee, pay, use, awesome, open, always, using, many, payment
Topic 6	Discount coupons and promotion policies	shopee, app, better, vouchers, free, keep, shipping, give, lot, thank
Topic 7	Shopping experience and platform reviews	shopping, great, online, platform, shopee, app, best, love, shop, easy
Topic 8	User friendliness and service quality	user, delivery, friendly, excellent, items, money, shopee, almost, door, value

Figure 3.11 Topic

3.6 Visualization and Interactive Presentation

To enhance the clarity and presentation of analytical results, this study employed various Python visualization libraries, including matplotlib, seaborn, wordcloud, and pyLDAvis, to generate the following outputs:

- **Bar Charts**: Distribution of sentiment based on user ratings and VADER analysis
- **Line Charts**: Monthly sentiment trend analysis
- **Word Clouds**: Keyword clouds for positive and negative reviews
- **Interactive Plot**: LDA topic visualization using pyLDAvis

Figure 3.12 presents the word cloud generated from Shopee user reviews labeled as “positive” based on their ratings. The most frequent keywords include *good*, *easy*, *delivery*, *fast*, *cheap*, *app*, *love*, and *shop*, indicating that users generally appreciate Shopee's ease of use, fast logistics, and pricing. The word cloud visually emphasizes these core aspects, with word size reflecting frequency, providing an intuitive overview of positive sentiment dimensions.

Figure 3.12 Positive Word Cloud

Figure 3.13 displays the word cloud generated from reviews labeled as “negative.” Prominent keywords such as *refund*, *customer*, *order*, *late*, *return*, *problem*, *crash*, and *bug* highlight recurring user concerns related to after-sales service, delivery delays, and system stability. This visualization offers valuable insight into the root causes of user dissatisfaction, enabling the platform to efficiently identify critical pain points and inform targeted improvements in product and service quality.

□3.3 □□□□□□□

Figure 3.13 Negative Word Cloud

Figure 3.14 presents the interactive LDA topic visualization generated using PyLDAvis. The scatterplot illustrates the relative distance and distribution density among the eight identified topics. The spatial separation of bubbles reflects semantic differences between topics, while the size of each bubble indicates the proportion of documents associated with that topic. By interacting with individual bubbles, users can explore the keyword distributions and weights associated with each topic, offering deeper insights into the focal points of user feedback.

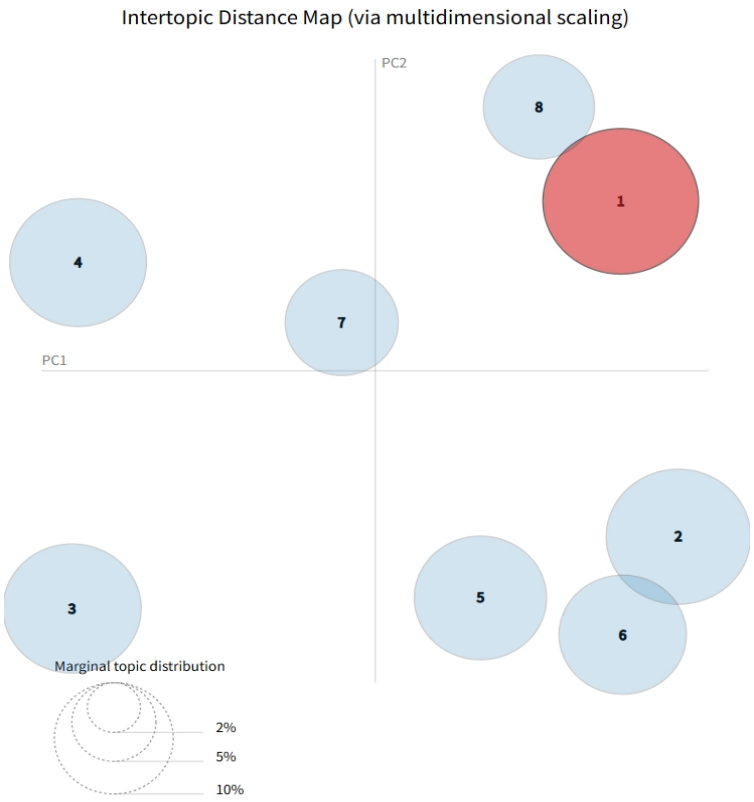


Figure 3.14 LDA Topic Visualization

Figure 3.15 illustrates the distribution of sentiment labels—positive, neutral, and negative—generated based on the user rating mapping. As shown in the figure, positive reviews constitute the majority, followed by negative reviews, with neutral feedback representing a smaller proportion. This sentiment distribution serves as a benchmark for subsequent supervised machine learning tasks in sentiment classification.

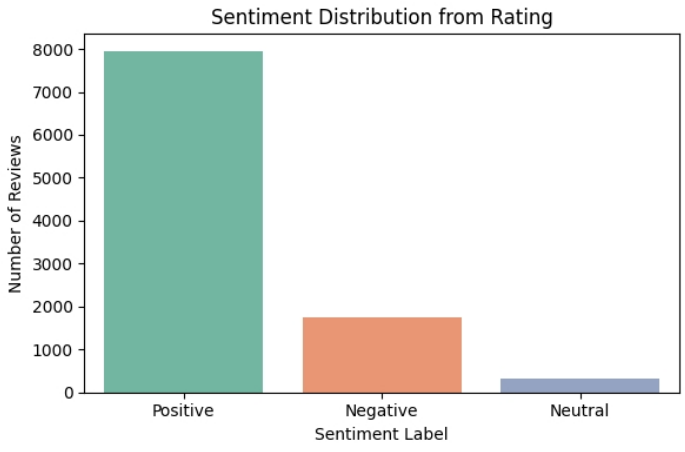


Figure 3.15 Distribution Of Sentiment Labels

Figure 3.16 shows the monthly trend of average user ratings in Shopee reviews (on a scale from 1 to 5). The overall trajectory reflects fluctuations in users' general satisfaction with the platform over time. For instance, temporary drops in ratings during certain promotional periods or system updates may indicate service bottlenecks or technical issues.

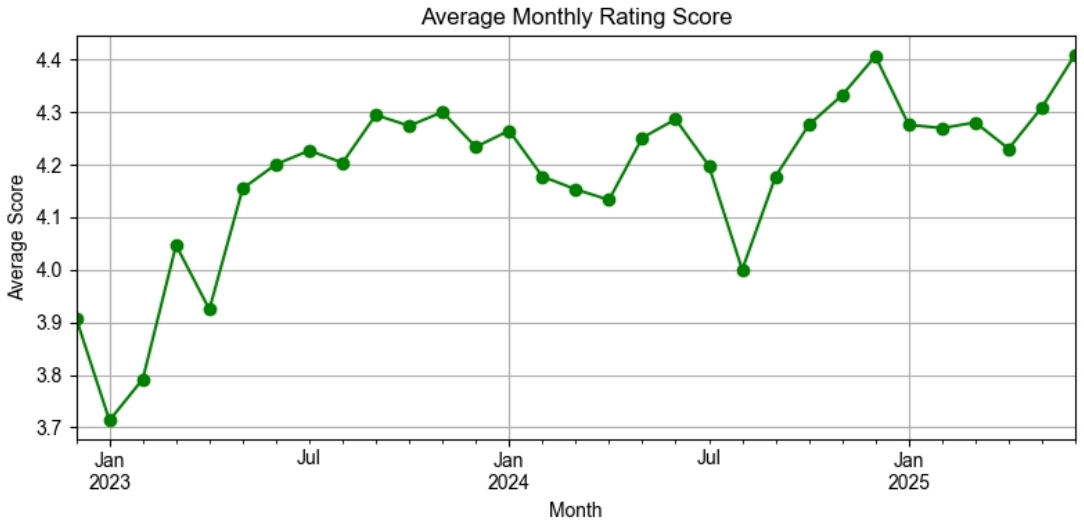


Figure 3.16 Monthly Trend Of Average User Ratings

Figure 3.17 illustrates the monthly proportion trends of sentiment labels (positive, negative, neutral) derived from user ratings. This chart helps reveal potential correlations between sentiment fluctuations and time (or operational events). For example, a sudden surge in negative sentiment in a particular month may correspond to a logistics disruption or an app crash incident.

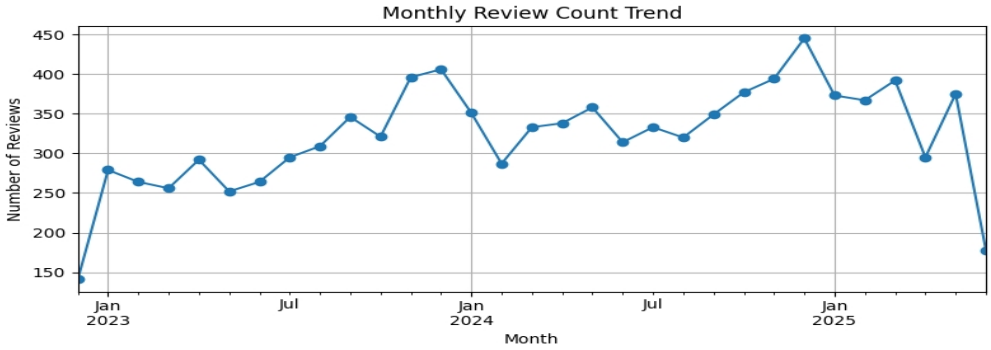


Figure 3.17 Monthly Proportion Trends Of Sentiment Labels

Chapter 4: Research Findings

4.1 Dataset Overview

4.1.1 DATA SOURCE AND FILTERING

The dataset consists of approximately 10,000 English-language user reviews of the Shopee app, scraped from the Google Play Store. To ensure data quality and analytical relevance, the following preprocessing steps were applied:

- Removal of null values and duplicate records;
- Filtering to retain only English reviews from the past five years;
- Cleaning and standardization of text fields for NLP tasks.

4.1.2 KEY VARIABLES AND SAMPLE CHARACTERISTICS

The cleaned dataset contains the following core variables:

- **reviewText**: Original user comment;
- **score**: User rating (1 to 5 stars);
- **Cleaned Review Text**: Standardized version of the review for NLP use;
- **Sentiment Label (from Rating)**: Sentiment category derived from the score;
- **VADER Sentiment / VADER Label**: VADER compound score and corresponding label;
- **Month**: Review date converted into a Year-Month format;
- **LDA Topic**: Assigned topic number (0–7) from LDA topic modeling.

The final dataset contains 9,809 valid reviews, with most entries posted between 2020 and 2024. The sample size is sufficiently representative for subsequent sentiment modeling and trend analysis.

4.1.3 Overview of Data Distribution

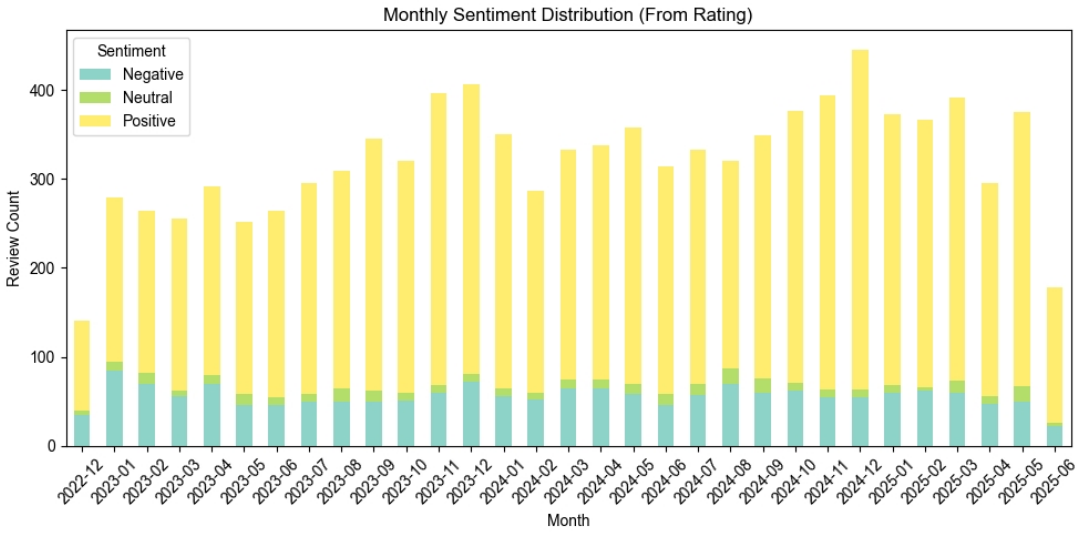


Figure 4.1 Monthly Sentiment Distribution(From Rating)

4.2 Sentiment Analysis Results (VADER)

4.2.1 VADER SCORING PERFORMANCE

To explore the sentiment distribution of user reviews, the VADER (Valence Aware Dictionary and sEntiment Reasoner) tool was applied to each review text. It calculates a **compound score** ranging from -1 to 1 to reflect overall sentiment polarity. Based on widely adopted thresholds, each review was categorized into one of three sentiment labels:

- **Positive**: compound score ≥ 0.05
- **Negative**: compound score ≤ -0.05
- **Neutral**: otherwise

The resulting sentiment distribution based on VADER scoring is summarized in the table below:

VADER Sentiment	Number of Comments	Percentage (%)
Positive	7202	72.02
Neutral	1395	13.95
Negative	1403	14.03

Figure 4.2 Sentiment Distribution Based On Vader Scoring

As shown in the table, positive sentiment dominates the review set, accounting for 72.02% of all comments. However, over 28% of the reviews reflect either neutral or negative emotions, indicating a notable level of user dissatisfaction that deserves attention from the platform.

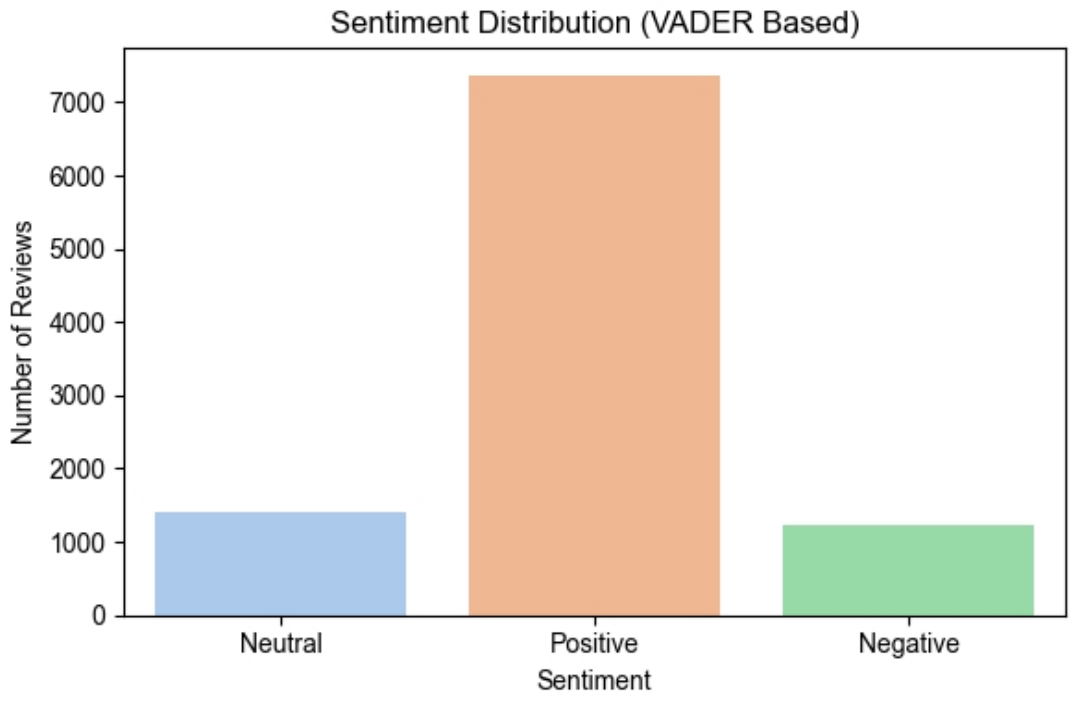


Figure 4.3 Sentiment Distribution(VADER Based)

4.2.2 MONTHLY SENTIMENT TREND COMPARISON

To examine how user sentiment evolves over time, the Review Date was converted into a standard "Year-Month" format. Then, monthly sentiment proportions were calculated based on two labeling approaches: one derived from user ratings ("Sentiment Label from Rating"), and the other from VADER sentiment analysis ("VADER Label"). The results are visualized in Figure 4.4 and Figure 4.5 respectively.

Figure 4.4 shows that sentiment labels derived from user ratings consistently indicate a dominant proportion of positive sentiment, with monthly shares exceeding 65%. Notably, the proportion of negative sentiment is relatively higher in the rating-based results when compared to those from VADER.

In contrast, Figure 4.5 presents the sentiment proportions as identified by the VADER tool, which shows a noticeably higher proportion of neutral sentiment. This suggests that many reviews may use neutral language even when assigned extreme ratings (e.g., 1 or 5 stars), highlighting the limitations of inferring emotion purely from ratings.

Overall, the proportion of **negative sentiment** is **higher in the rating-based results** than in the VADER analysis, while **neutral sentiment** is more prominent in the **VADER-based results**. This indicates that relying solely on ratings might **overestimate sentiment polarity**, whereas VADER provides a more nuanced and text-driven understanding of user sentiment.

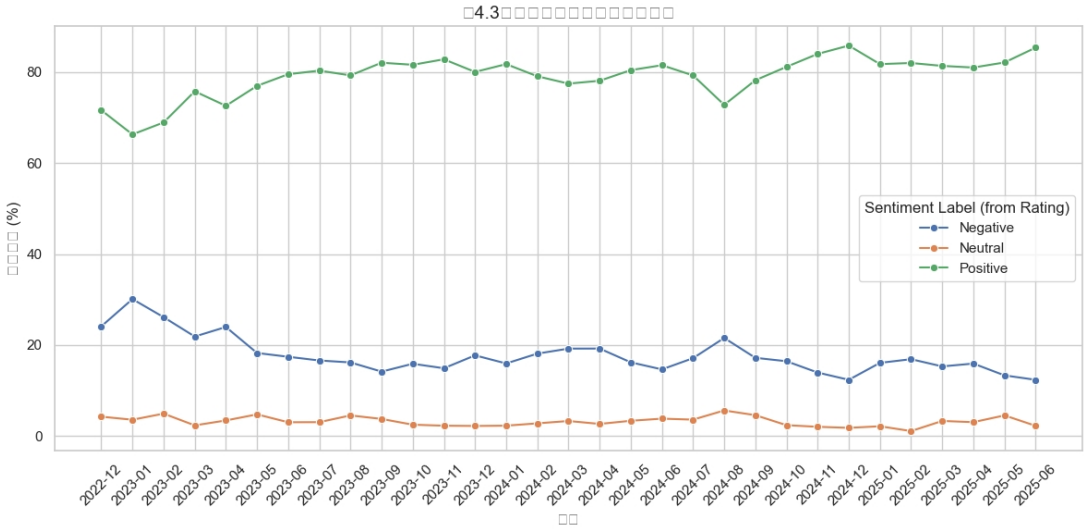


Figure 4.4 Rating_Trend

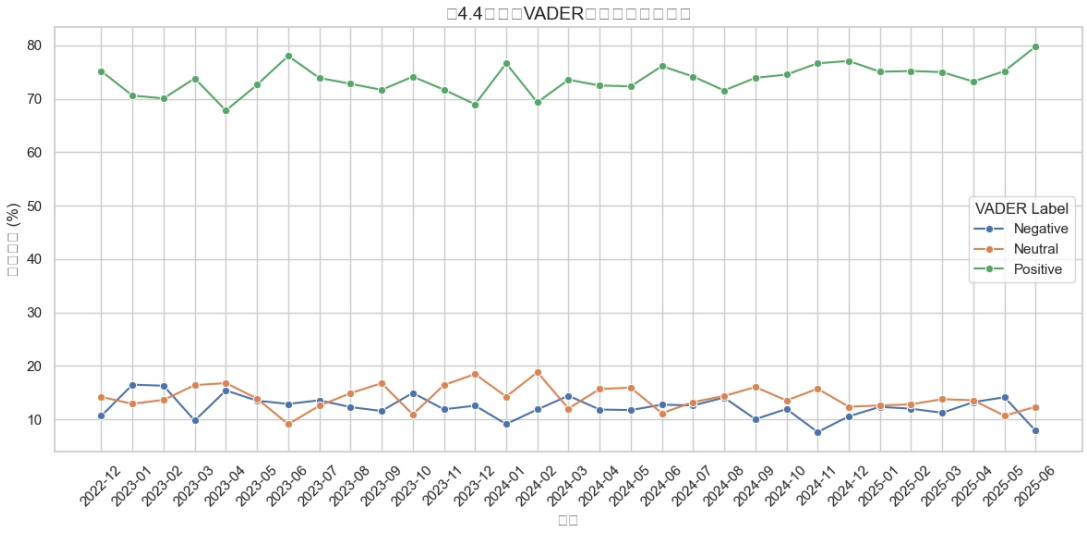


Figure 4.5 Vader_Trend

4.3 Topic Modeling Results (LDA)

4.3.1 TOPIC KEYWORDS AND DISTRIBUTION

Using Gensim, an LDA model was constructed with the number of topics set to eight. The top 10 keywords for each topic are listed below, representing the core semantic content of each cluster:

Number	Topic Name	Top 10 Keywords
Topic 1	Problems with the APP usage	app, shopee, back, account, review, phone, still, use, even, get
Topic 2	After-sales and logistics services	refund, shopee, delivery, service, item, customer, time, seller, return, order
Topic 3	Product content and price evaluation	products, nice, product, sellers, price, shopee, like, variety, items, prices
Topic 4	Cost-effectiveness and user satisfaction	good, easy, delivery, fast, use, service, far, happy, shopee, app
Topic 5	Payment and Account Experience	app, shopee, pay, use, awesome, open, always, using, many, payment
Topic 6	Discount coupons and promotion policies	shopee, app, better, vouchers, free, keep, shipping, give, lot, thank
Topic 7	Shopping experience and platform reviews	shopping, great, online, platform, shopee, app, best, love, shop, easy
Topic 8	User friendliness and service quality	user, delivery, friendly, excellent, items, money, shopee, almost, door, value

Figure 4.6 Top 10 Keywords

This keyword distribution allows us to interpret each topic's thematic focus. For example:

- **Topic 1** focuses on account and login issues;
- **Topic 2** centers around refunds, delivery problems, and customer service;
- **Topic 3** relates to general product quality and pricing;
- **Topic 4** highlights positive shopping experiences such as "easy," "good," and "happy."

These topics offer a high-level overview of user concerns and satisfaction points across Shopee's app reviews.

4.3.2 WORD CLOUDS OF POSITIVE AND NEGATIVE REVIEWS

The Cleaned Review Text was split into two groups based on sentiment labels (positive and negative), and separate word clouds were generated for each.

In the **positive reviews**, frequently occurring words include *"easy"*, *"love"*, *"delivery"*, and *"shopping"*, reflecting user appreciation of convenience, platform usability, and shipping efficiency.

In contrast, **negative reviews** often contain terms such as *"late"*, *"refund"*, *"crash"*, *"bad"*, and *"poor"*, indicating user dissatisfaction primarily related to delivery delays and app performance issues.

□3.2 □□□□□□□

Figure 4.7 Positive Word Cloud

□3.3 □□□□□□□

Figure 4.7 Negetive Word Cloud

These word clouds help visually highlight key concerns and satisfaction drivers within user feedback.

4.3.3 LDA TOPIC VISUALIZATION (INTERACTIVE PLOT)

To better illustrate how topics are distributed across user reviews, this study utilized the pyLDAvis tool to generate an interactive visualization (see Figure 3.14). The resulting two-dimensional scatter plot presents each topic's relative importance (represented by circle size) and semantic similarity (reflected by their spatial proximity).

Each circle in the plot corresponds to a topic identified by the LDA model. The area of each circle is proportional to the frequency of that topic within the entire corpus. The distance between circles indicates the degree of dissimilarity between topics—the closer the circles, the more similar the topics' word compositions; the farther apart, the more distinct their semantics.

From Figure 3.14, we observe that Topic 2 (related to *refunds and customer service*) and Topic 4 (focused on *delivery experience*) appear relatively large and close to each other, suggesting that these themes occur frequently and share a number of overlapping terms. Meanwhile, Topic 7 (about *overall shopping experience*) is positioned more independently, indicating that user discussions on this topic are more distinct and cohesive.

This interactive visualization provides an intuitive lens to understand user concerns and lays the groundwork for the subsequent sentiment–topic joint analysis.

4.3.4 SENTIMENT–TOPIC CROSS ANALYSIS

By combining each review's assigned LDA Topic with its corresponding VADER Sentiment Label, a cross-tabulation was created to explore the sentiment distribution across topics.

Key observations from the crosstab include:

- **Topic 2** shows the **highest proportion of negative sentiment (35.8%)**, primarily linked to issues related to **returns, customer service, and delivery**.
- **Topic 4** and **Topic 7** exhibit extremely **high proportions of positive sentiment**, reaching **81.95%** and **86.17%**, respectively. These are associated with **shopping convenience** and **overall platform experience**, indicating high user satisfaction.
- **Neutral sentiment** is mainly concentrated in **Topic 6** and **Topic 5**, which may relate to **payment processes** or **app usability and stability**.

This joint analysis highlights how specific themes elicit varying emotional responses, offering valuable guidance for targeted service improvements.

LDA Topic	Positive (%)	Neutral (%)	Negative (%)
Topic 1	71.85	13.42	14.73
Topic 2	50.62	13.58	35.8
Topic 3	75.44	14.78	9.78
Topic 4	81.95	10.17	7.88
Topic 5	73.91	13.92	12.17
Topic 6	66.23	17.54	16.23
Topic 7	86.17	7.16	6.67
Topic 8	76.42	13.1	10.48

Figure 4.8 Topic Table

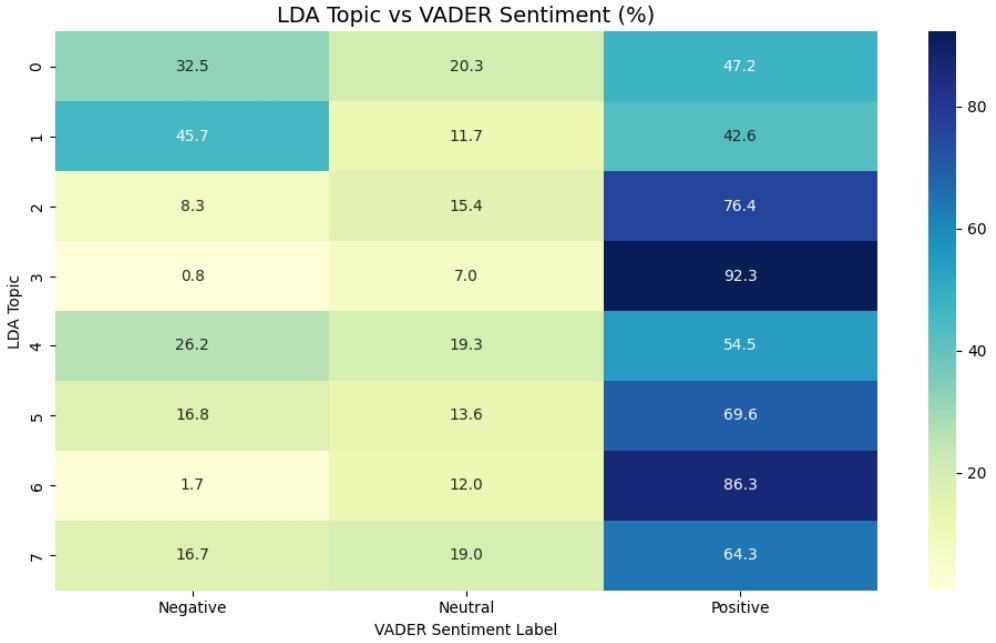


Figure 4.9 LDA Topic vs Sentiment(%)

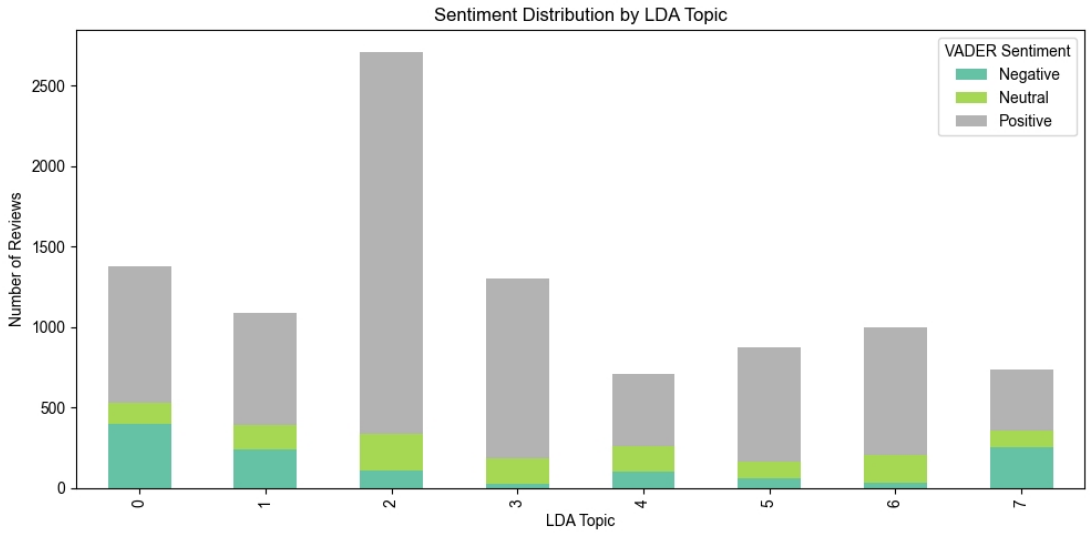


Figure 4.10 Sentiment Distribution by LDA Topic

4.4 Summary

This chapter applied sentiment analysis and LDA topic modeling on a dataset of 10,000 Shopee user reviews. The key findings are as follows:

- Overall, user sentiment is predominantly positive, though issues related to logistics and app performance frequently trigger negative feedback.
- Sentiment results generated by VADER largely align with those inferred from user ratings.
- The eight extracted topics cover key dimensions such as user experience, delivery service, and payment security.
- A clear relationship was observed between certain topics and specific sentiment trends, offering actionable insights for Shopee's product and service optimization.

Chapter 5: Discussion

5.1 Insights from Sentiment Analysis

Using both VADER and rating-based labels, this study revealed that while the overall sentiment of Shopee reviews was predominantly positive, negative sentiment was concentrated around specific issues such as delivery delays and app crashes. VADER demonstrated high accuracy in identifying

positive and negative sentiment but was less effective in detecting neutral tones—likely due to its lexicon-based limitations.

5.2 Model Comparison and Methodological Reflection

As a rule-based tool, VADER offers advantages in deployment and speed, making it well-suited for short, informal reviews. However, it struggles with context-heavy or sarcastic content. In contrast, LDA topic modeling uncovers the latent semantic structure of reviews but lacks inherent sentiment orientation. When combined, these methods provide a balanced approach to both structural and emotional insights.

5.3 Behavioral Insights and Operational Implications

The eight topics extracted via LDA span key areas such as shopping experience, app performance, pricing, and logistics. Notably, Topic 2 (technical issues) and Topic 4 (delivery problems) showed high concentrations of negative sentiment, indicating areas requiring urgent improvement. In contrast, positive feedback centered around usability and affordability, suggesting these remain Shopee's competitive strengths.

5.4 Practical Applications

The findings can directly inform product management, customer experience optimization, and sentiment monitoring systems within Shopee. It is recommended that the platform develop an automated dashboard to track new reviews daily using integrated LDA and VADER analyses for proactive strategy adjustments.

5.5 Research Contributions and Limitations

This study presents a hybrid approach that combines lexicon-based sentiment analysis and topic modeling to extract structured insights from mobile app reviews. Key limitations include:

- Absence of machine learning classifiers for deeper sentiment modeling;
- Focus solely on English reviews, excluding multilingual user feedback;
- Lack of behavioral data integration for multimodal analysis.

Chapter 6: Conclusion

6.1 Main Findings

This study developed a comprehensive analytical pipeline for Shopee user reviews, including data collection, preprocessing, sentiment classification, topic modeling, and visualization. Key findings include:

- Most reviews express positive sentiment, but certain issues—especially in logistics and technical performance—trigger negative responses;
- VADER effectively classifies sentiment and, when combined with LDA, helps trace emotional responses to specific themes;
- Keyword and topic analysis provides actionable directions for product and service improvements.

6.2 Limitations

- Data was limited to publicly available reviews and lacked behavioral or transaction-related metadata;
- Sentiment analysis relied on a lexicon-based method without applying deep learning models;
- Some topics were semantically mixed, suggesting room for parameter tuning in LDA modeling.

6.3 Future Research Directions

- Incorporate transformer-based models (e.g., BERT) for more nuanced sentiment analysis;
- Explore multimodal modeling by integrating unstructured reviews with structured purchase data;
- Design response strategies to manage critical themes with high negative sentiment, contributing to an effective reputation management framework.

References

1.Aras, S., Yusuf, M., Ruimassa, R., Agustinus, E., & Palalangan, E. (2024). *Sentiment Analysis on Shopee Product Reviews Using IndoBERT*. **Journal of Information Systems and Informatics**, 6(3), 1616–1626. en.wikipedia.org+11researchgate.net+11journal-isi.org+11

2.Sentiment Analysis using Machine Learning Models on Shopee Reviews. (2023). *Journal of Social Science and Modern Research*, 14(2). aasmr.org

3.*The Shopee Application User Reviews Sentiment Analysis Employing Naïve Bayes Algorithm*. (2023). *International Journal of Software Engineering and Computer Science (IJSECS)*, 3(3), 194–204. pdfs.semanticscholar.org

4.*Sentiment Analysis of User Reviews of E-commerce Applications: Case Study on the Shopee Platform*. (2024). *Journal of Social Science and Modern Research*, 5(4), 986–…。 jurnal.unai.edu+6pdfs.semanticscholar.org+6jsss.co.id+6

5.Nguyen, T. M. (2023). *Topic Modelling and Sentiment Analysis of Customer Reviews for B2C E-commerce Platforms in Vietnam: A Comparative Study of Lazada, Shopee, Tiki, and Sendo* (Undergraduate thesis). The College of Wooster. openworks.wooster.edu

6.*Sentiment Analysis on Product Reviews from Shopee Marketplace using the Naive Bayes Classifier*. (2022). *AASMR Journal of Statistics & Management Research*, 11(3). researchgate.net+5researchgate.net+5pdfs.semanticscholar.org+5

7.*Topic Modelling and Aspect-Based Sentiment Analysis in Shopee Instagram Comments*. (2024). *Social Science & Modern Research*, Vol.5(4). pdfs.semanticscholar.org+1openlibrary.telkomuniversity.ac.id+1

8.Telkom University. (2018). *Analisa Konten Media Sosial e-Commerce pada Instagram menggunakan Sentiment Analysis dan LDA-Based Topic Modeling: Studi kasus Shopee Indonesia*. (Unpublished report). researchgate.net+2openlibrary.telkomuniversity.ac.id+2pdfs

APPENDIX A.1: KEY PYTHON CODE SNIPPETS

• Data Acquisition

```python
from google_play_scraper import reviews, Sort
import pandas as pd

def fetch_reviews(app_package_name, n_reviews=10000, lang='en', country='us'):  1 个用法
    all_reviews = []
    cursor = None
    while len(all_reviews) < n_reviews:
        revs, cursor = reviews(
            app_package_name,
            lang=lang,
            country=country,
            sort=Sort.NEWEST,
            count=200,
            continuation_token=cursor
        )
        all_reviews.extend(revs)
        if not cursor:
            break
    return all_reviews[:n_reviews]

package_name = "com.shopee.sg"

print("Starting to scrape Shopee reviews...")
data = fetch_reviews(package_name, n_reviews=10000)

df = pd.DataFrame([{
    'reviewId': r['reviewId'],
    'userName': r['userName'],
    'userImage': r['userImage'],
    'reviewText': r['content'],
    'score': r['score'],
    'thumbsUpCount': r['thumbsUpCount'],
    'reviewCreatedVersion': r.get('reviewCreatedVersion', ''),
    'reviewDate': r['at'],
    'replyContent': r.get('replyContent', ''),
    'repliedAt': r.get('repliedAt', ''),
} for r in data])

df.to_csv('shopee_play_reviews.csv', index=False, encoding='utf-8-sig')
print(f"✅ Successfully captured and saved {len(df)} comments to shopee_play_reviews.csv")
```

• Data Cleaning Function: clean_text()

```python
import pandas as pd
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.sentiment.vader import SentimentIntensityAnalyzer
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('vader_lexicon')
df = pd.read_csv(r"D:\YOUHUAN\USM Course\web and social media#ABM503\assignment\OPTION1\data\shopee_play_reviews.csv")
def clean_text(text):  1 个用法
    text = str(text).lower()
    text = re.sub( pattern: r"http\S+|www\S+",  repl: "", text)
    text = re.sub( pattern: r"[^a-z\s]",  repl: " ", text)
    tokens = text.split()
    stop_words = set(stopwords.words('english'))
    tokens = [w for w in tokens if w not in stop_words and len(w) > 2]
    return " ".join(tokens)
df['Cleaned Review Text'] = df['reviewText'].apply(clean_text)
def label_from_score(score):  1 个用法
    if score >= 4:
        return 'Positive'
    elif score == 3:
        return 'Neutral'
    else:
        return 'Negative'
df['Sentiment Label (from Rating)'] = df['score'].apply(label_from_score)
df['Month'] = pd.to_datetime(df['reviewDate']).dt.to_period('M')
sia = SentimentIntensityAnalyzer()
df['VADER Sentiment'] = df['Cleaned Review Text'].apply(lambda x: sia.polarity_scores(x)['compound'])
def vader_to_label(score):  1 个用法
    if score >= 0.05:
        return 'Positive'
    elif score <= -0.05:
        return 'Negative'
    else:
        return 'Neutral'
df['VADER Label'] = df['VADER Sentiment'].apply(vader_to_label)
df.to_csv("shopee_reviews_cleaned.csv", index=False, encoding='utf-8-sig')
print("✅ Data cleaning is complete and has been saved as shopee_reviews_cleaned.csv")
```

• LDA Modeling and Visualization Script (gensim, pyLDAvis)

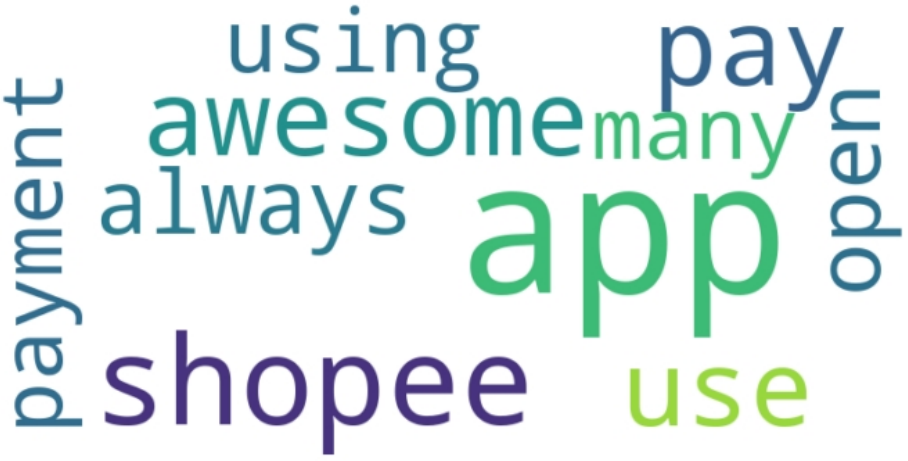
```python
import pandas as pd
import gensim
from gensim import corpora
from wordcloud import WordCloud
import matplotlib.pyplot as plt
import pyLDAvis.gensim_models

df = pd.read_csv("shopee_reviews_cleaned.csv")

texts = df['Cleaned Review Text'].dropna().apply(lambda x: x.split()).tolist()
dictionary = corpora.Dictionary(texts)
corpus = [dictionary.doc2bow(text) for text in texts]

lda_model = gensim.models.ldamodel.LdaModel(corpus=corpus,
                                            id2word=dictionary,
                                            num_topics=8,
                                            random_state=42,
                                            passes=10,
                                            per_word_topics=True)

print("\n📌 Top Words per Topic:")
for i, topic in lda_model.show_topics(formatted=False):
    words = [word for word, _ in topic]
    print(f"🟣 Topic {i + 1}: {', '.join(words)}")

for i, topic in lda_model.show_topics(formatted=False):
    word_freq = {word: weight for word, weight in topic}
    wc = WordCloud(width=800, height=400, background_color='white').generate_from_frequencies(word_freq)
    plt.figure(figsize=(8, 4))
    plt.imshow(wc, interpolation='bilinear')
    plt.axis("off")
    plt.title(f"Topic {i + 1} WordCloud")
    plt.tight_layout()
    plt.savefig(f"topic_{i + 1}_wordcloud.png")
    plt.close()
print("✅ All the word clouds have been created!")
print("📊 Generating the interaction diagram... Please wait...")
import os
import tempfile
os.environ['JOBLIB_TEMP_FOLDER'] = tempfile.mkdtemp(prefix="lda_temp_", dir="C:/Temp")
vis = pyLDAvis.gensim_models.prepare(lda_model, corpus, dictionary)
pyLDAvis.save_html(vis, "shopee_lda_gensim_visualization.html")
print("✅ The visual graph has been saved as shopee_lda_gensim_visualization.html")
```

APPENDIX A.2: MODEL OUTPUT TABLES AND CHARTS

Topic 7 WordCloud

Topic 6 WordCloud

Topic 5 WordCloud



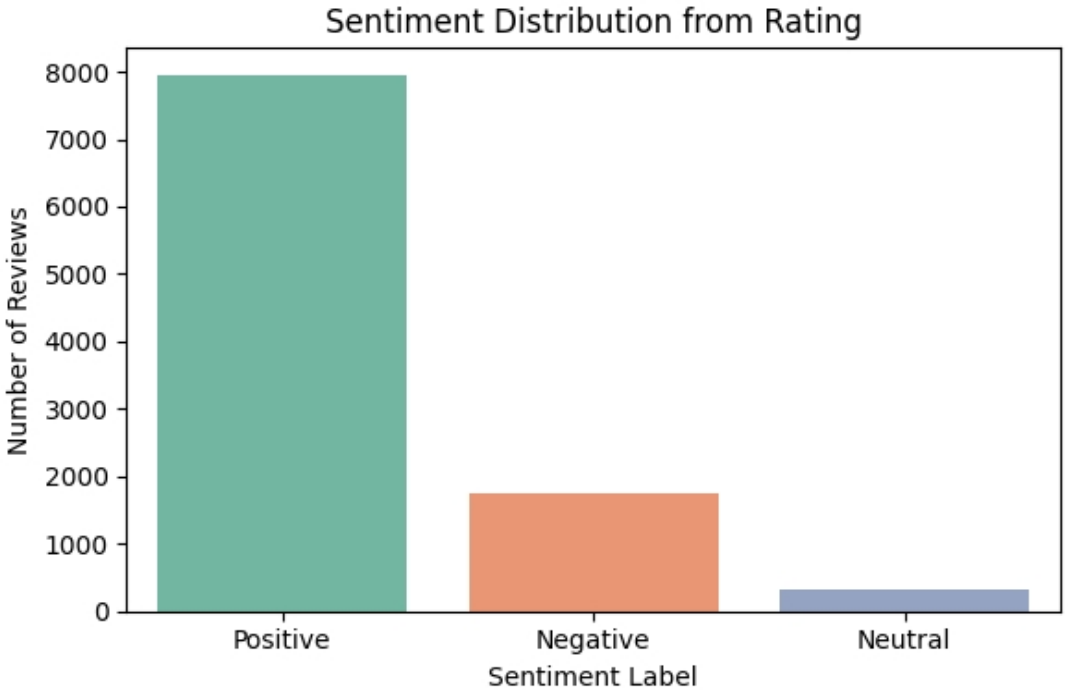
Topic 4 WordCloud

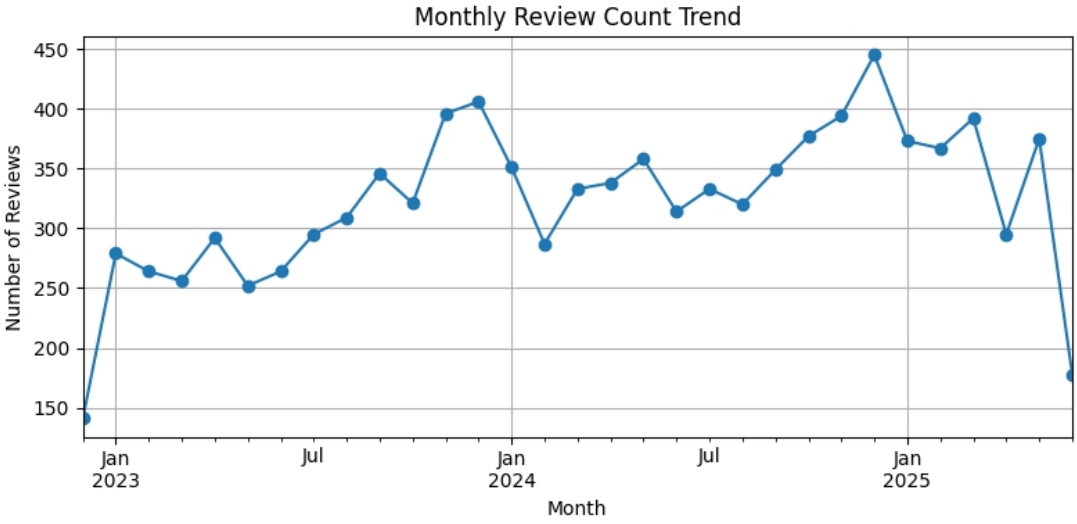
Topic 3 WordCloud

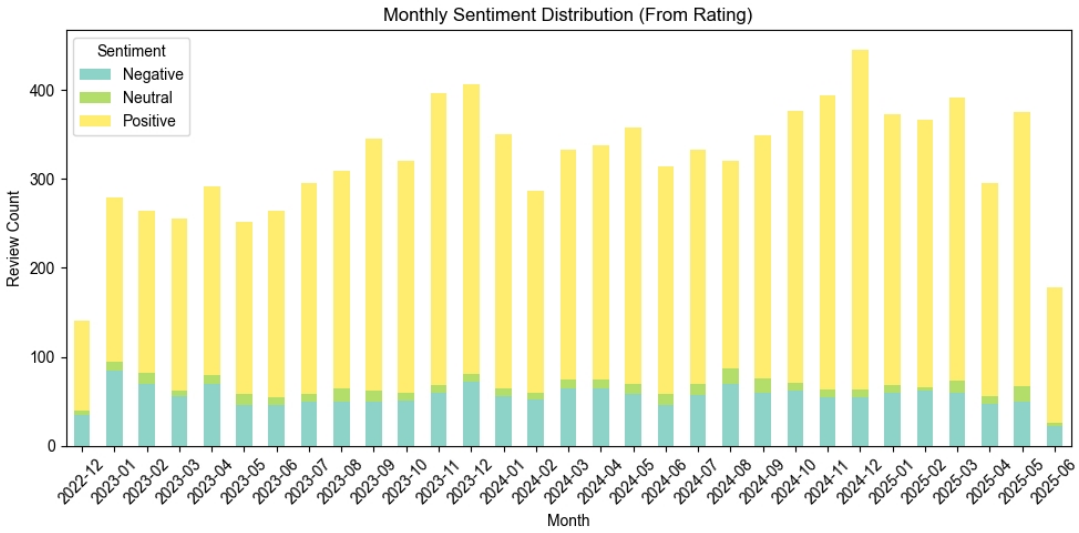
Topic 2 WordCloud

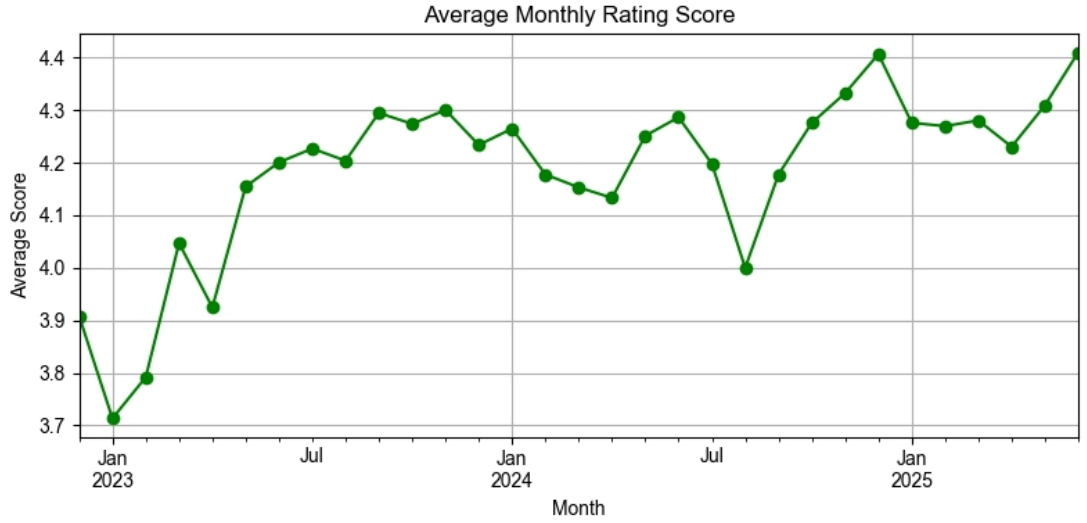
Topic 1 WordCloud

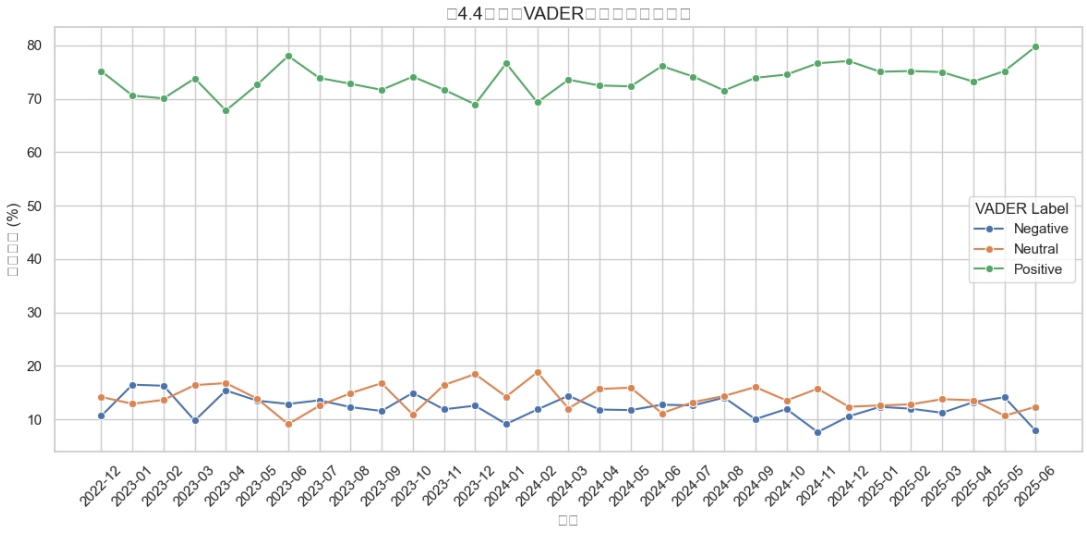
Topic 8 WordCloud


Sentiment Distribution from Rating
Number of Reviews
8000
7000
6000
5000
4000
3000
2000
1000
0
Positive
Negative
Neutral
Sentiment Label

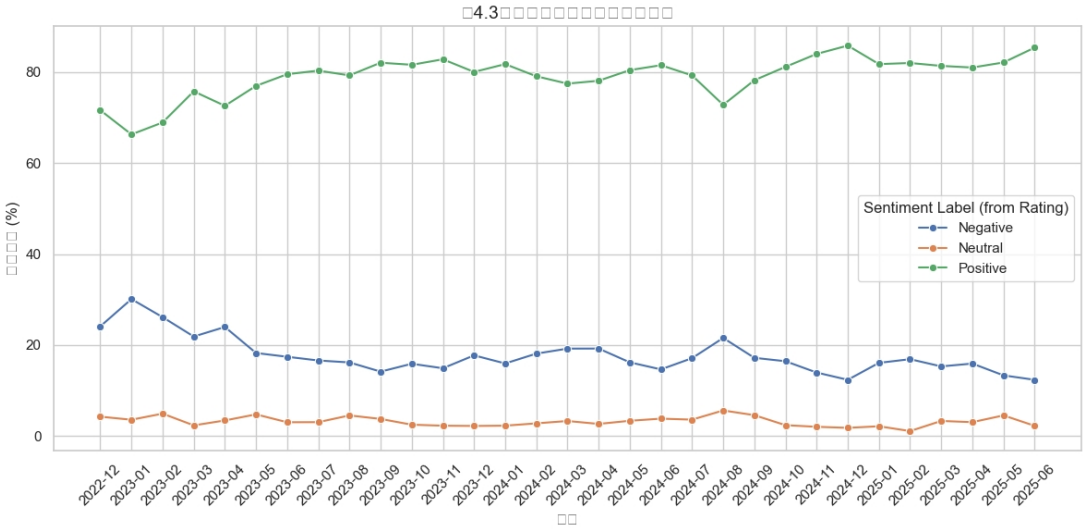

Monthly Review Count Trend
Number of Reviews
450
400
350
300
250
200
150
Jan
2023
Jul
Jan
2024
Jul
Jan
2025
Month

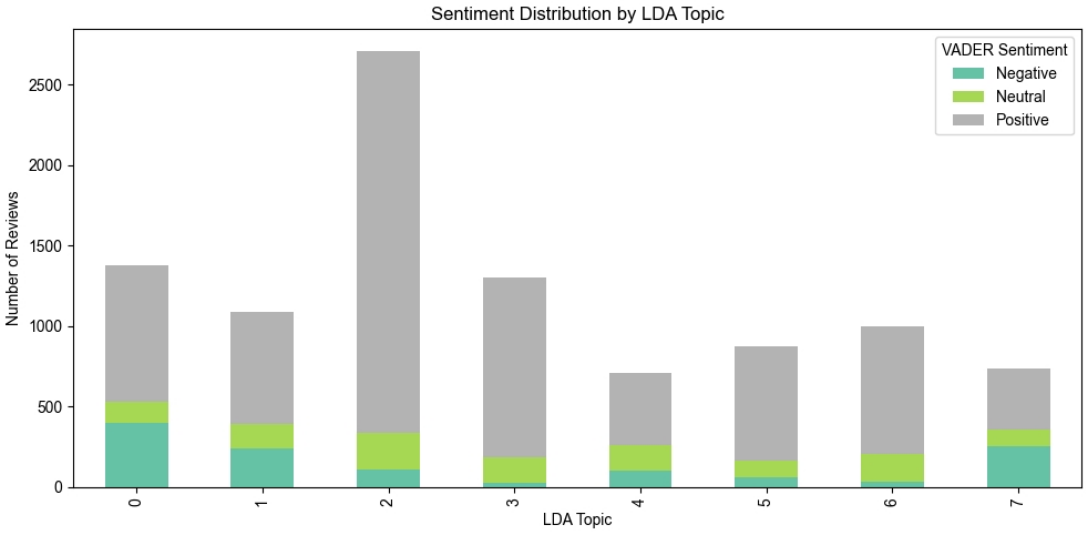

Monthly Sentiment Distribution (From Rating)
Sentiment
Negative
Neutral
Positive
Review Count
Month

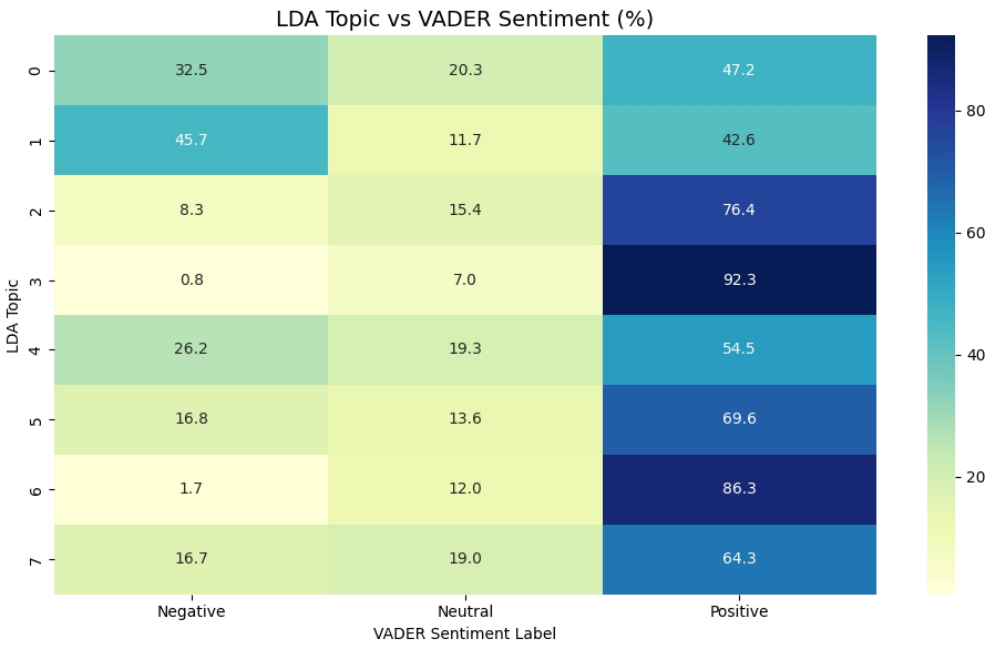

Average Monthly Rating Score
Average Score
Month
Jan
2023
Jul
Jan
2024
Jul
Jan
2025

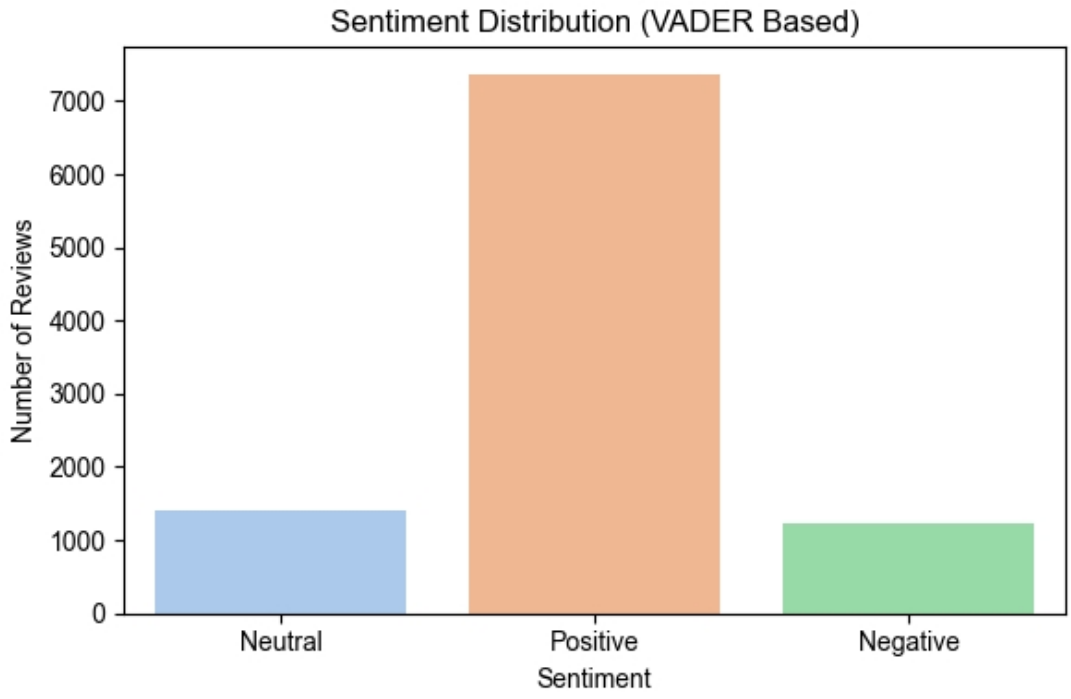

□4.4□□□VADER□□□□□□□□
VADER Label
Negative
Neutral
Positive

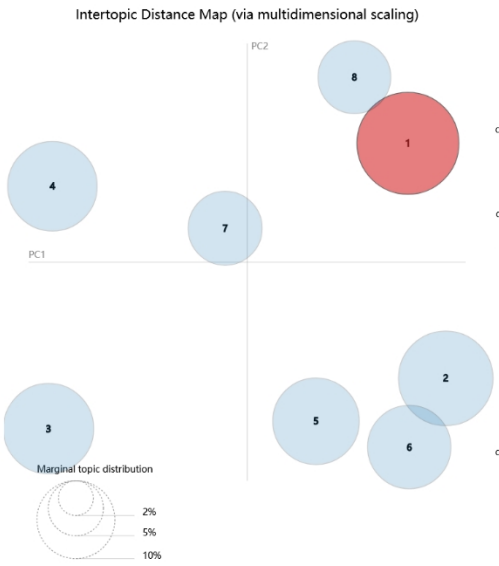
□4.3□□□□□□□□□□□□□□

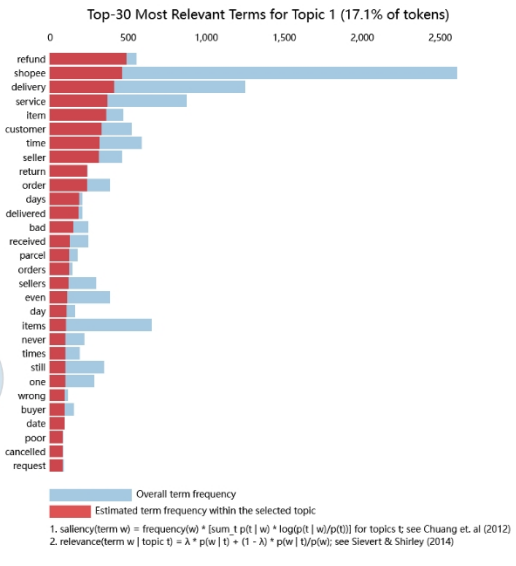

□□□□ (%)
Sentiment Label (from Rating)
Negative
Neutral
Positive
□□


Sentiment Distribution by LDA Topic
VADER Sentiment
Negative
Neutral
Positive
Number of Reviews
LDA Topic


LDA Topic vs VADER Sentiment (%)
LDA Topic
0
1
2
3
4
5
6
7
32.5
20.3
47.2
45.7
11.7
42.6
8.3
15.4
76.4
0.8
7.0
92.3
26.2
19.3
54.5
16.8
13.6
69.6
1.7
12.0
86.3
16.7
19.0
64.3
Negative
Neutral
Positive
VADER Sentiment Label
80
60
40
20


Sentiment Distribution (VADER Based)
Number of Reviews
7000
6000
5000
4000
3000
2000
1000
0
Neutral
Positive
Negative
Sentiment


Intertopic Distance Map (via multidimensional scaling)
PC2
PC1
8
1
4
7
2
3
5
6
Marginal topic distribution
2%
5%
10%


Top-30 Most Relevant Terms for Topic 1 (17.1% of tokens)
0
500
1,000
1,500
2,000
2,500
refund
shopee
delivery
service
item
customer
time
seller
return
order
days
delivered
bad
received
parcel
orders
sellers
even
day
items
never
times
still
one
wrong
buyer
date
poor
cancelled
request
Overall term frequency
Estimated term frequency within the selected topic
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)