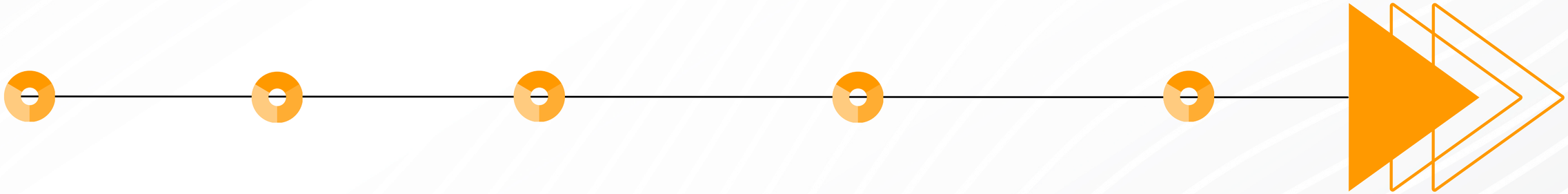
AMAZON REVIEW ANALYSIS

BANA 212: DATA & PROGRAMMING ANALYTICS

PRESENTED BY SHIVANI VALLAMDAS, CHRISTIE SHIN, GEMA ZHU, VISHAL SRIVASTAVA, SHUAI ZHAO

TABLE OF CONTENTS



Background

Overview of our topic and goal

Data Collection & Cleaning

Data Manipulation & Cleaning Methods

Text & Sentiment Analysis

Textual & Sentiment Analysis of Reviews

ML Models

K-Means Clustering & Decision Tree

Conclusion

Conclusive Analysis of Data & Recommendations

BACKGROUND

Problem: Amazon is susceptible to many fake reviews. Customers lose trust in Amazon if fake reviews are not removed consistently.

Why Amazon: One of the biggest e-commerce sites with hundreds and thousands of reviews for products.

Process: Use text & sentiment analysis, k-means clustering, & decision trees to find suspicious reviews and user behavior

RESEARCH FOCUS

Can we **detect suspicious or potentially fake Amazon reviews** using text & sentiment analysis & ML methods from reviews and user behaviour?

01 Do suspicious reviewers display abnormal posting patterns (frequency, timing)?

02 Do suspicious reviews exhibit more extreme sentiment (overly positive or overly negative) than legitimate reviews?

03 Do suspicious reviews show abnormal linguistic patterns, such as overly generic wording, repeated phrases, or unusually short/long text?

Kaggle Dataset Titled: “Amazon Products Review”

DATA SOURCE

Total Records: 568454

Total Columns: 10

DATA CLEANING

Removed Duplicates

- Deleted 1,208 duplicate reviews based on ProductID, UserId, Time, and Text
- Final dataset: 567,246 unique reviews

Helpfulness Check

- Removed 2 invalid rows (Numerator > Denominator)

Timestamp Conversion

- Converted Unix time → Readable Dates (ConvertedDate)

Missing Values

- Checked Summary, Text, ProfileName → none found

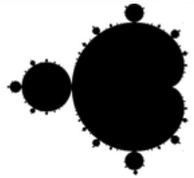
Text Cleaning

- Standardized CleanedText using TRIM + LOWER (remove capital and extra spaces)

Feature Engineering

- Added key metrics:
  - ReviewLength
  - UserReviewCount
  - SingleDayReviewFrequency
  - AverageScorePerUser
  - DailyReviewRate
  - Countof5StarReviews

```
----Sentiment analysis----
----Summaries----
100%|██████████| 567244/567244 [01:22<00:00, 6908.86it/s]
Summary Sentiment calculated.
----Cleaned Text----
100%|██████████| 567244/567244 [04:32<00:00, 2084.51it/s]
Text Sentiment calculated.

Sample Results:
   Score                Summary  SummarySentiment  \
0      5  Good Quality Dog Food               0.7
1      1      Not as Advertised               0.0
2      4  "Delight" says it all               0.0
3      2         Cough Medicine               0.0
4      5            Great taffy               0.8

                                         CleanedText  TextSentiment
0  i have bought several of the vitality canned d...       0.450000
1  product arrived labeled as jumbo salted peanut...      -0.033333
2  this is a confection that has been around a fe...       0.133571
3  if you are looking for the secret ingredient i...       0.166667
4  great taffy at a great price. there was a wide...       0.483333
```

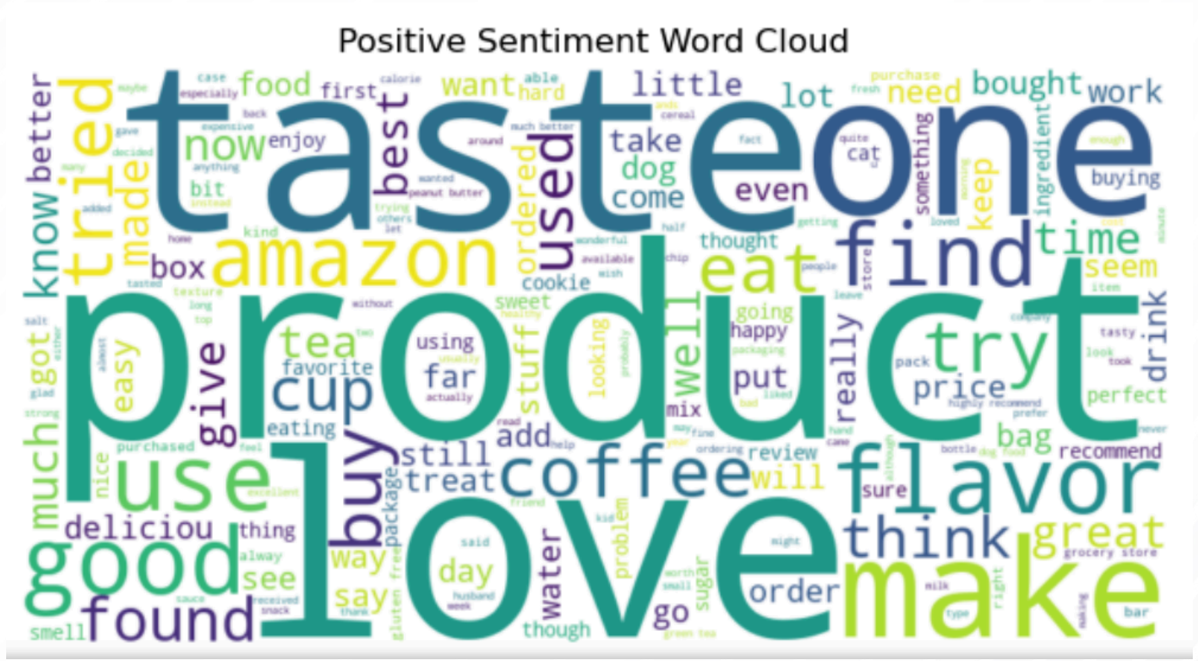


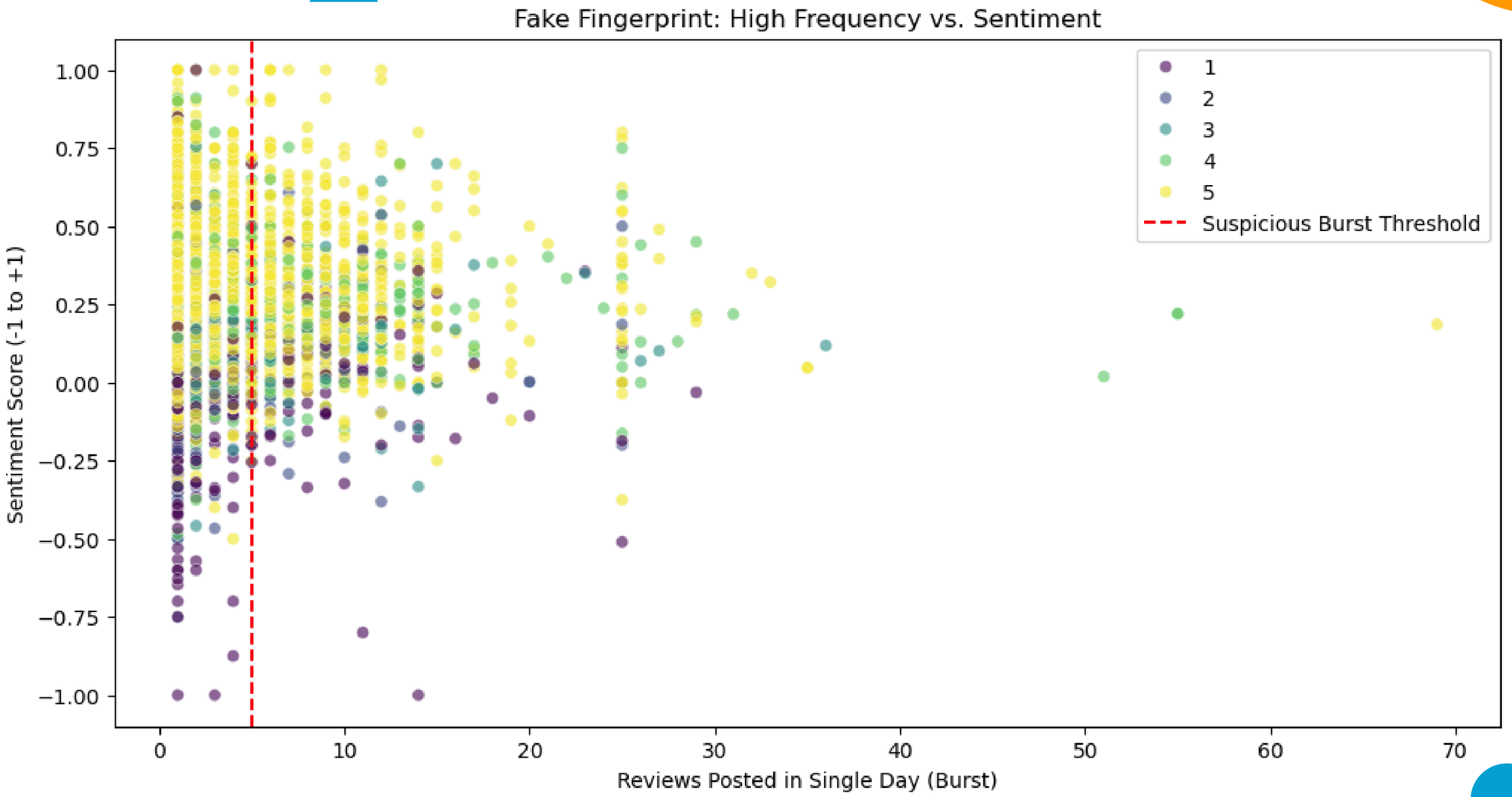
TextBlob

TextBlob is a Python library for processing textual data. It provides a consistent API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, and more.

TEXT & SENTIMENT ANALYSIS

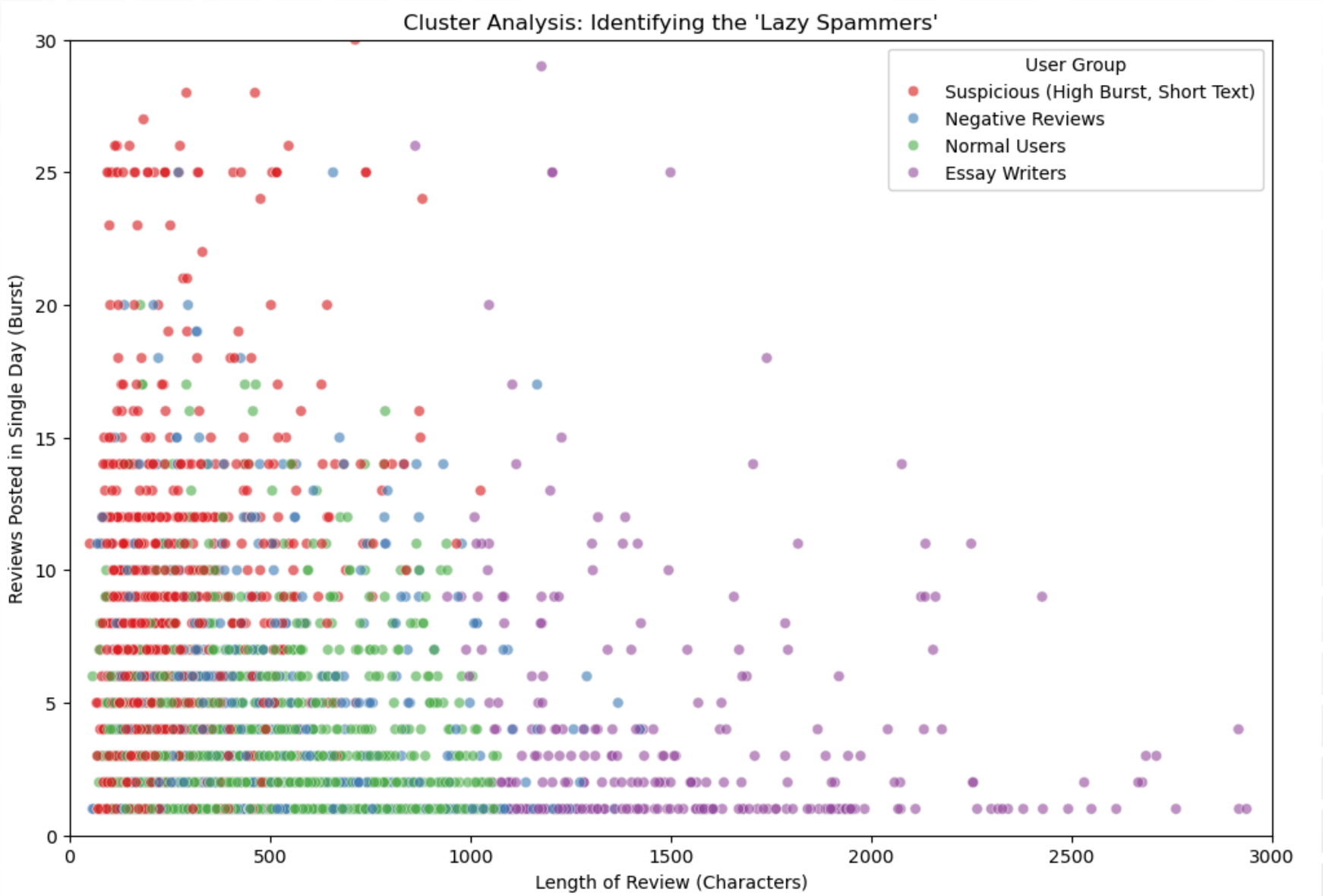
Text & Sentiment Analysis

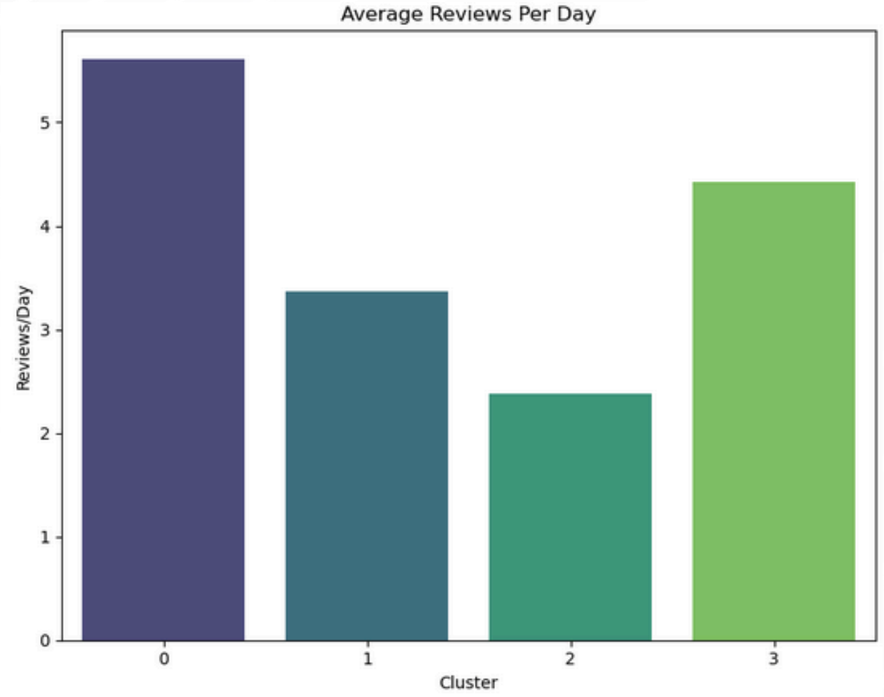


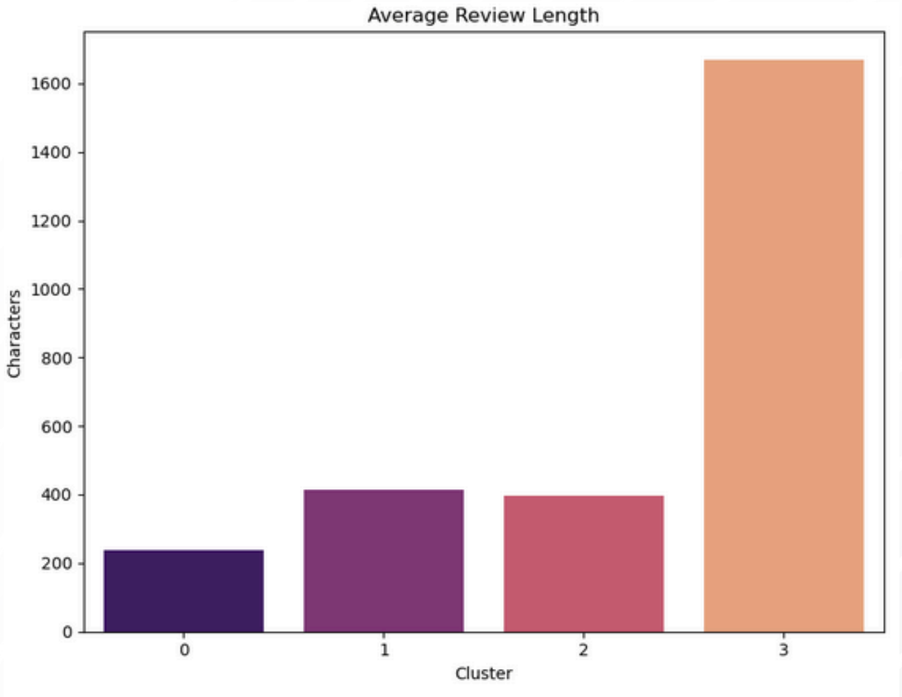

Fake Fingerprint: High Frequency vs. Sentiment
Sentiment Score (-1 to +1)
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
10
20
30
40
50
60
70
Reviews Posted in Single Day (Burst)
1
2
3
4
5
Suspicious Burst Threshold

K-MEANS CLUSTERING

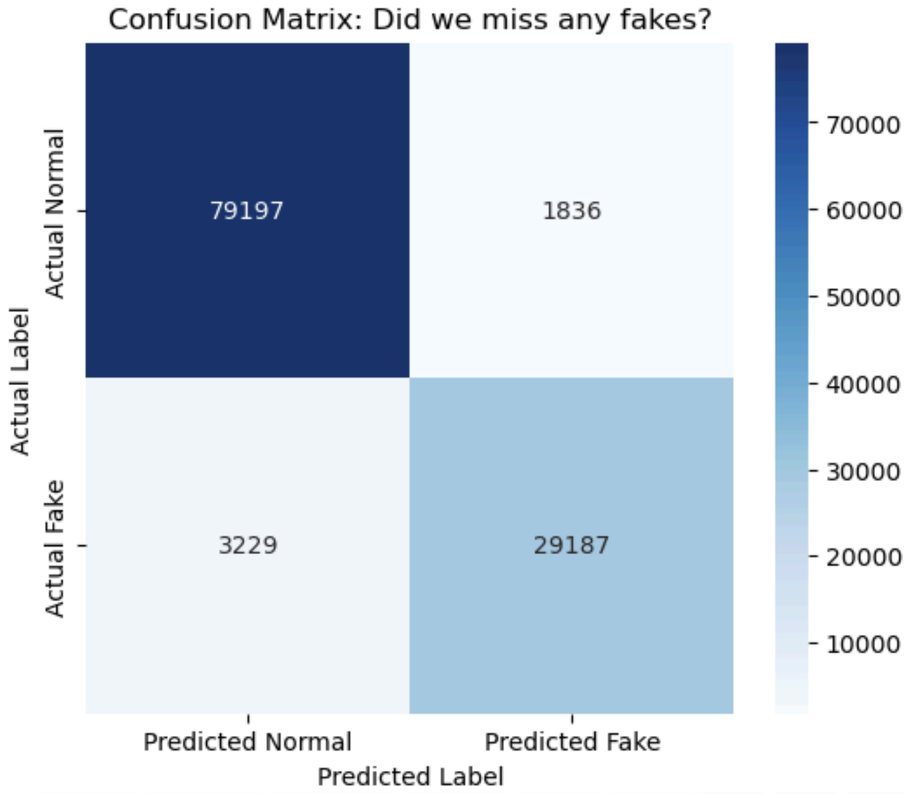
```
----K Means Clustering----
Clustering Complete!

----CLUSTER PROFILES----
         SingleDayReviewFrequency  TextSentiment  ReviewLength     Score  \
Cluster
0                        5.604724       0.480741    236.200080  4.796454
1                        3.371013       0.038884    412.425066  1.806346
2                        2.383607       0.188697    397.564530  4.770088
3                        4.424158       0.151587   1666.990326  4.194939

           Count
Cluster
0         162155
1         106654
2         262670
3          35765
```


Cluster Analysis: Identifying the 'Lazy Spammers'
User Group
Suspicious (High Burst, Short Text)
Negative Reviews
Normal Users
Essay Writers
Reviews Posted in Single Day (Burst)
Length of Review (Characters)
0
5
10
15
20
25
30
0
500
1000
1500
2000
2500
3000





K-MEANS CLUSTERING

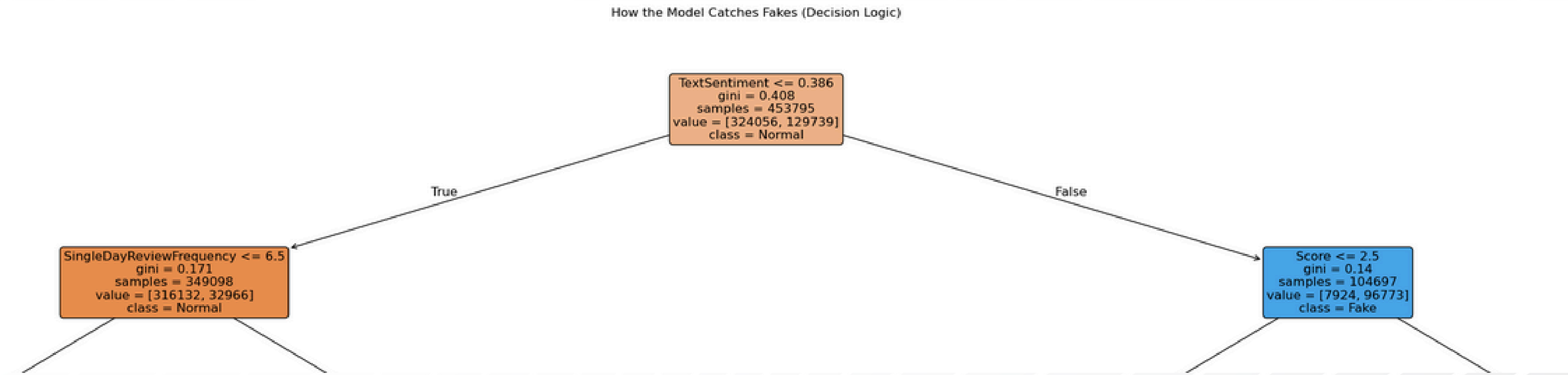


Decision Tree Accuracy: 95.54%

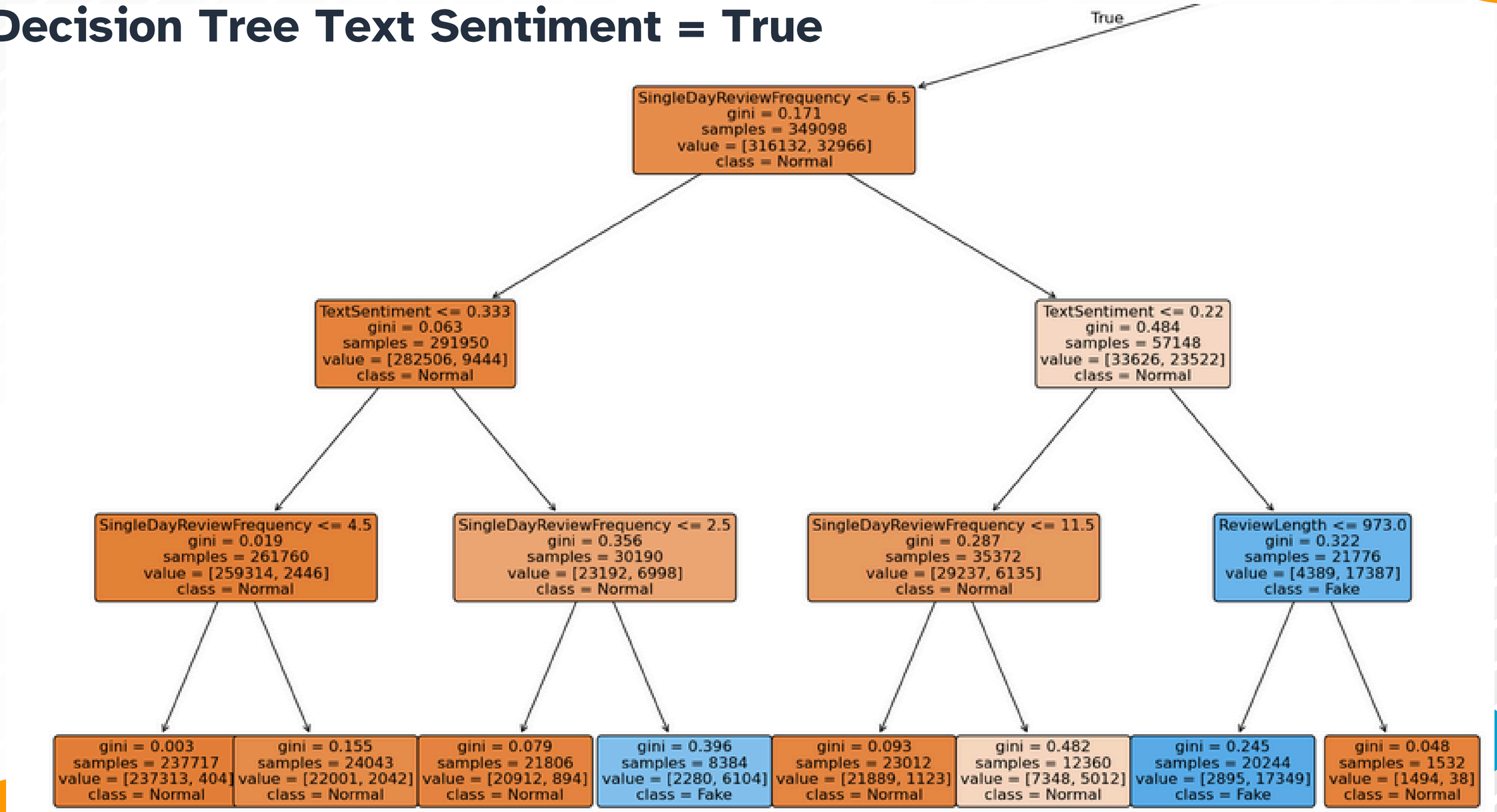
	precision	recall	f1-score	support
Normal	0.96	0.98	0.97	81033
Fake	0.94	0.90	0.92	32416
accuracy			0.96	113449
macro avg	0.95	0.94	0.94	113449
weighted avg	0.96	0.96	0.96	113449

DECISION TREES

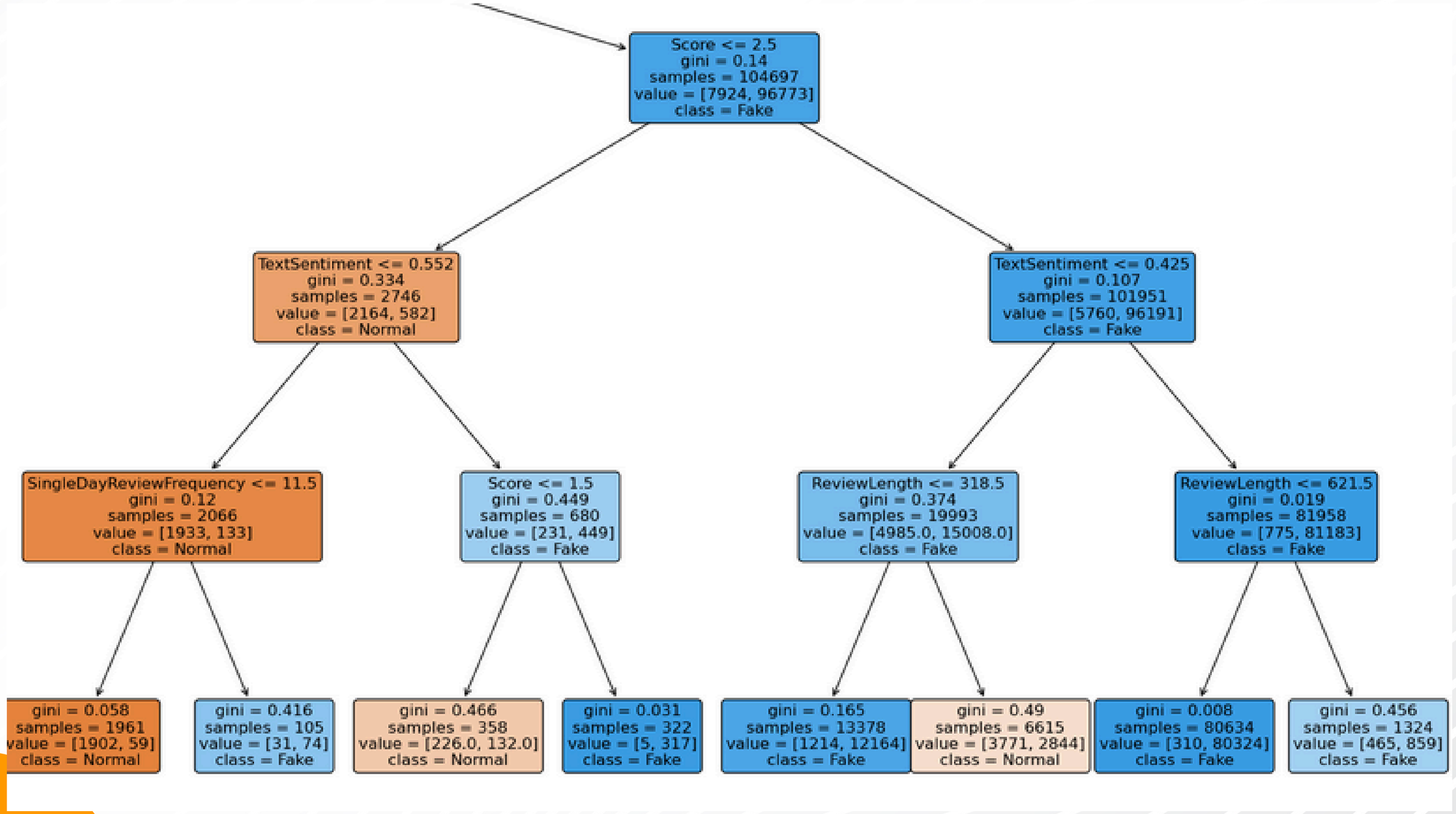
Decision Tree Head



Decision Tree Text Sentiment = True

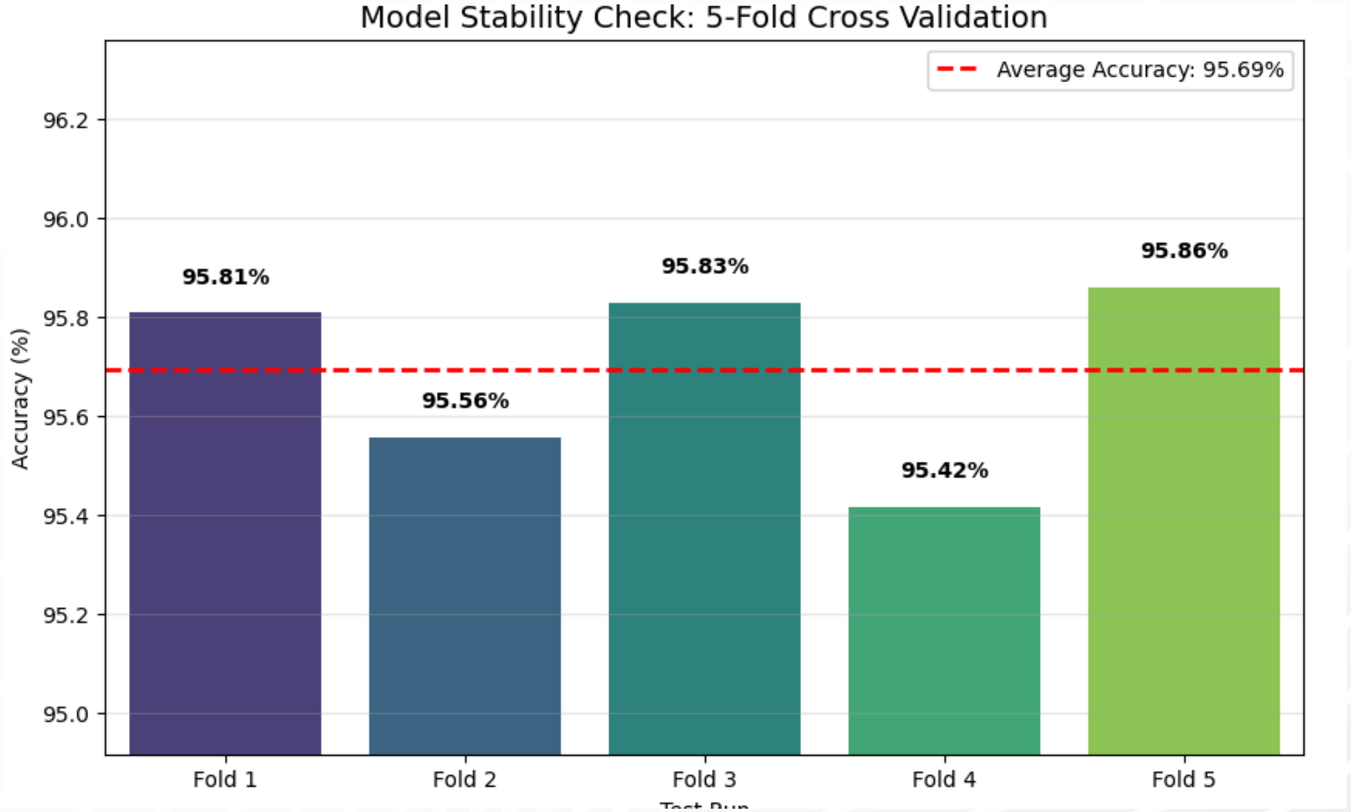


Decision Tree Text Sentiment = False



5-Fold Cross Validation

```
Results:
Scores for each run: [0.95807808 0.95556594 0.95828081 0.95416443 0.95858896]
Average Accuracy:    95.69%
Standard Deviation:  0.176
```



RESULTS

- Unsupervised clustering phase flagged **~28.5% of the total dataset (162,155 reviews)** as exhibiting anomalous, high-risk behavior

- **Suspicious reviews averaged 236 characters in length**, approximately 40% shorter than the 397-character average of legitimate users

- **141 specific review scripts (>50 characters)** that were posted largely unchanged by multiple different user accounts, providing undeniable evidence of coordinated review farms.

- Decision Tree achieved **95.54% accuracy → fraudulent behavior follows strict detectable rules**

- Decision Tree model identified **6.5 reviews per day** as critical tipping point for fraud; activity above this level is statistically predictive of bot ‘bursts’.

RECOMMENDATIONS

1. Implement Automated “Burst-Activity” Fraud & Sentiment Mismatch Flags

a. Flag reviews where emotional tone in the text contradicts the user’s numeric rating (e.g., a 5-star review with negative language)
b. Temporarily freeze posting ability for those who posts 7+ reviews per day
c. Require CAPTCHA or identity verification for repeated high-volume or inconsistent sentiment activity

2. Develop a Reviewer Risk Scoring System

a. Profile Reviewer Behavior
b. Track user level metrics like posting frequency, sentiment consistency, and text-length variation to identify unusual patterns.
c. Assign Credibility Scores
   - Generate a Reviewer Credibility Score combining behavioral and text-based feature, and use it to prioritize which reviews need manual verification.

LIMITATIONS

- Reviews are from around 2010s → model detects old bot attacks but **does not account for modern threats** like LLM-generated text (e.g., ChatGPT)

- Lacked verified "Fake/Real" tags from Amazon → **model predicts mathematical anomalies (high-risk behavior)** rather than confirmed fraud convictions

- Detection is **optimized for high-volume "burst" attacks and lazy bots**

- Strict threshold **(>6.5 reviews/day) flags all high-activity users**

THANK YOU!

PRESENTED BY **SHIVANI VALLAMDAS, CHRISTIE SHIN, GEMA ZHU, VISHAL SRIVASTAVA, SHUAI ZHAO**

BANA 212: DATA & PROGAMMING ANALYTICS