Technical Report: A2A, MCP, and ML for Sentiment Analysis Agents

1. Introduction

The growing need to classify user opinions from unstructured text data such as tweets and product reviews has driven the development of automated sentiment analysis systems. This technical report outlines a scalable, cloud-native architecture built on A2A (Agent-to-Agent) communication, a centralized Multi-Agent Control Plane (MCP), and integrated Machine Learning (ML) components. The system supports Twitter and iPhone sentiment analysis via modular agents managed through a shared coordination framework.

2. Goals

- Develop an intelligent, modular system that classifies sentiment across multiple sources.
- Enable scalability and low-latency inference through distributed deployment.
- Empower agents to operate independently while routing intelligently via a central orchestrator.
- Deliver a fully deployable system on Docker/Kubernetes cloud infrastructure.

3. Problem Statement

Organizations face challenges in:

- Extracting sentiment from large volumes of social and review data.
- Maintaining multiple ML services without centralized control.
- Ensuring integration and routing among distributed intelligent agents. This system solves these problems using MCP, agent routing, and automated deployment pipelines.

4. Dataset Description and EDA Analysis

iPhone Review Dataset

- Source: Amazon product reviews
- Fields: reviewDescription, ratingScore
- Label derivation: Rating 1–2 → Negative, 3 → Neutral, 4–5 → Positive

Key Findings from Initial EDA

1. Rating Distribution

- The majority of reviews have a **rating of 5**, suggesting a generally positive reception.
- Ratings of **1 and 2** are less common, indicating relatively fewer dissatisfied customers.

2. Verification Status

- **Verified reviews dominate** the dataset, contributing to over 90% of total entries.
- Verified reviews tend to be **more positive**, confirming their reliability.

3. Country-wise Review Volume

- **India** is the most represented country, followed by the **United States**.
- Countries like **Japan**, **UAE**, and **Egypt** also appear but contribute fewer reviews.

4. Temporal Trends

- A clear **increase in review volume over time**, peaking in mid-2024.
- This may align with product launches or seasonal sales (e.g., festive seasons).

5. Sentiment Analysis

- Based on a rating-based classification:
  - **Positive sentiment (rating ≥ 4)** dominates.
  - **Negative and neutral sentiments** are relatively rare.

Deeper Insights from Advanced Visuals

6. Word Cloud of Review Titles

- Words like **"iPhone," "good," "phone," "battery,"** and **"product"** are highly frequent.
- Indicates focus on general quality and specific features.

7. Monthly Average Ratings

- Monthly ratings **fluctuate** but stay mostly positive.
- Sudden dips or spikes can indicate issues like **product defects**, **patch releases**, or **viral feedback**.

8. Verified vs Unverified Ratings

- **Verified reviews consistently rate higher** than unverified ones.
- This is further confirmed by bar charts showing average scores across verification status.

9. Stacked Bar of Verified Reviews per Rating

- Across all ratings, the **proportion of verified reviews is high**.
- Interestingly, even lower ratings have a significant number of verified purchases, implying **real dissatisfaction** among genuine buyers.

TF-IDF Insights

10. Top Words in Positive Reviews

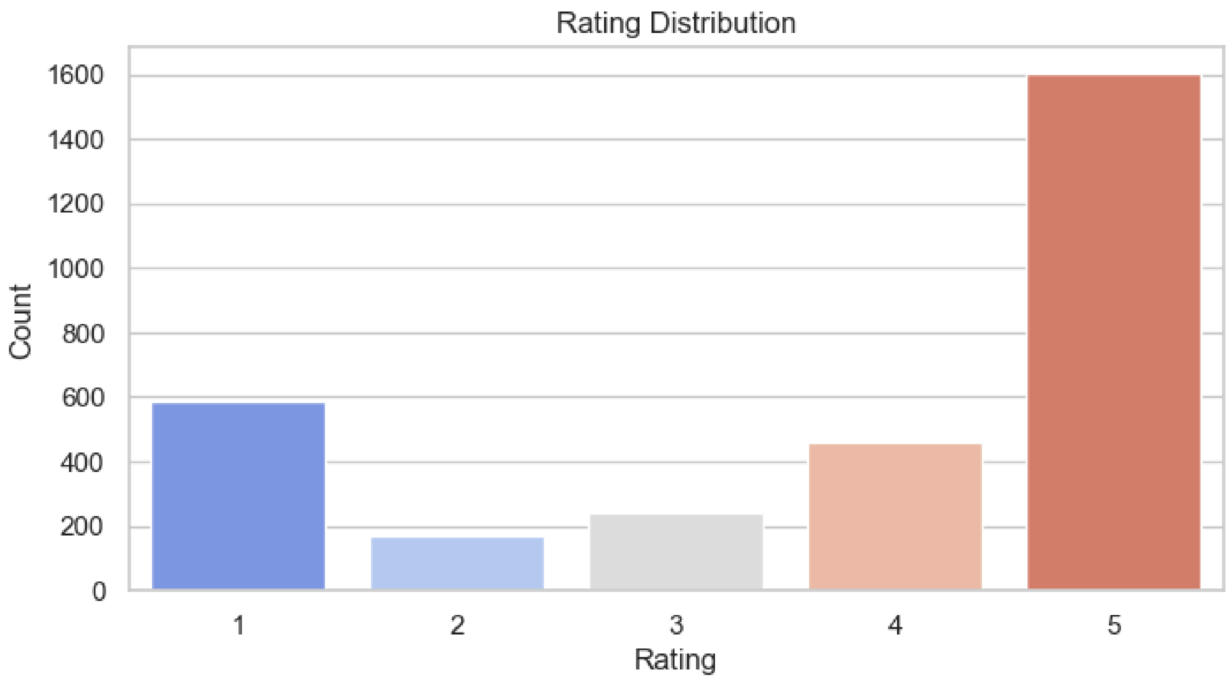
- Common terms: **"good," "phone," "product," "camera," "quality," "nice"**
- Customers praise aspects like performance, battery, camera, and build quality.

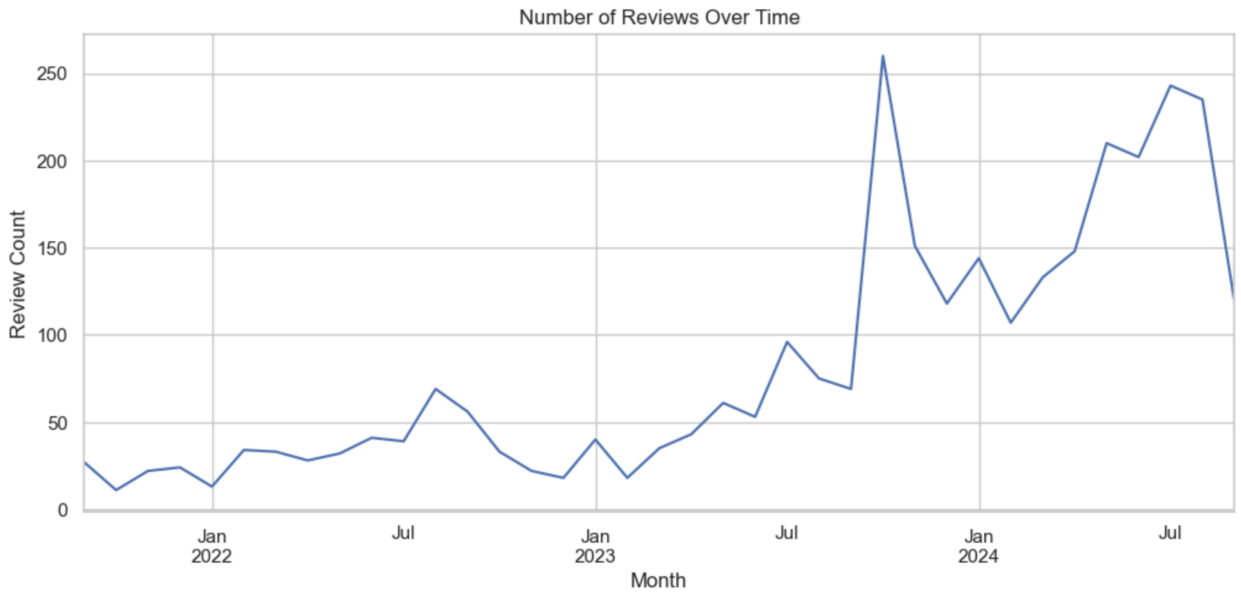
11. Top Words in Negative Reviews

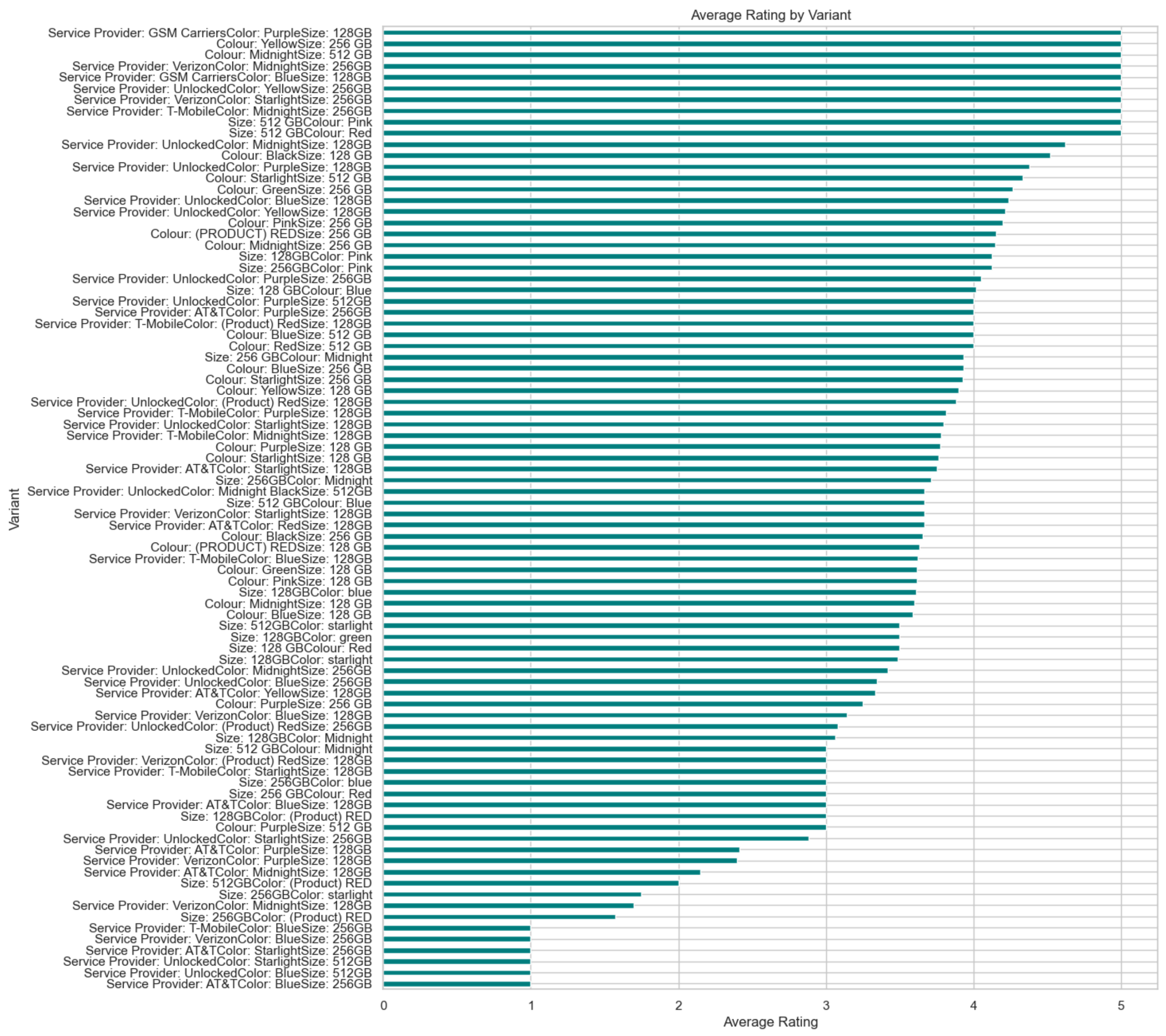
- Common terms: **"bad," "battery," "poor," "worst," "disappointed"**
- Battery and device quality are also the top **sources of negative sentiment**, showing they're double-edged factors.

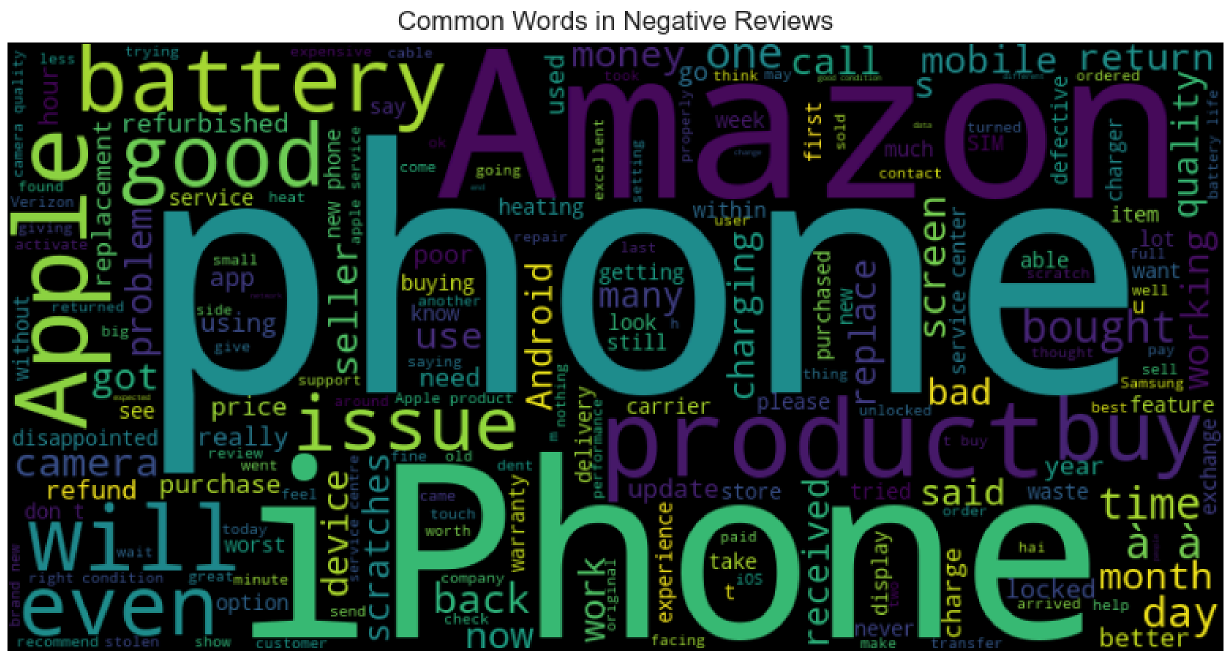
Final Observations

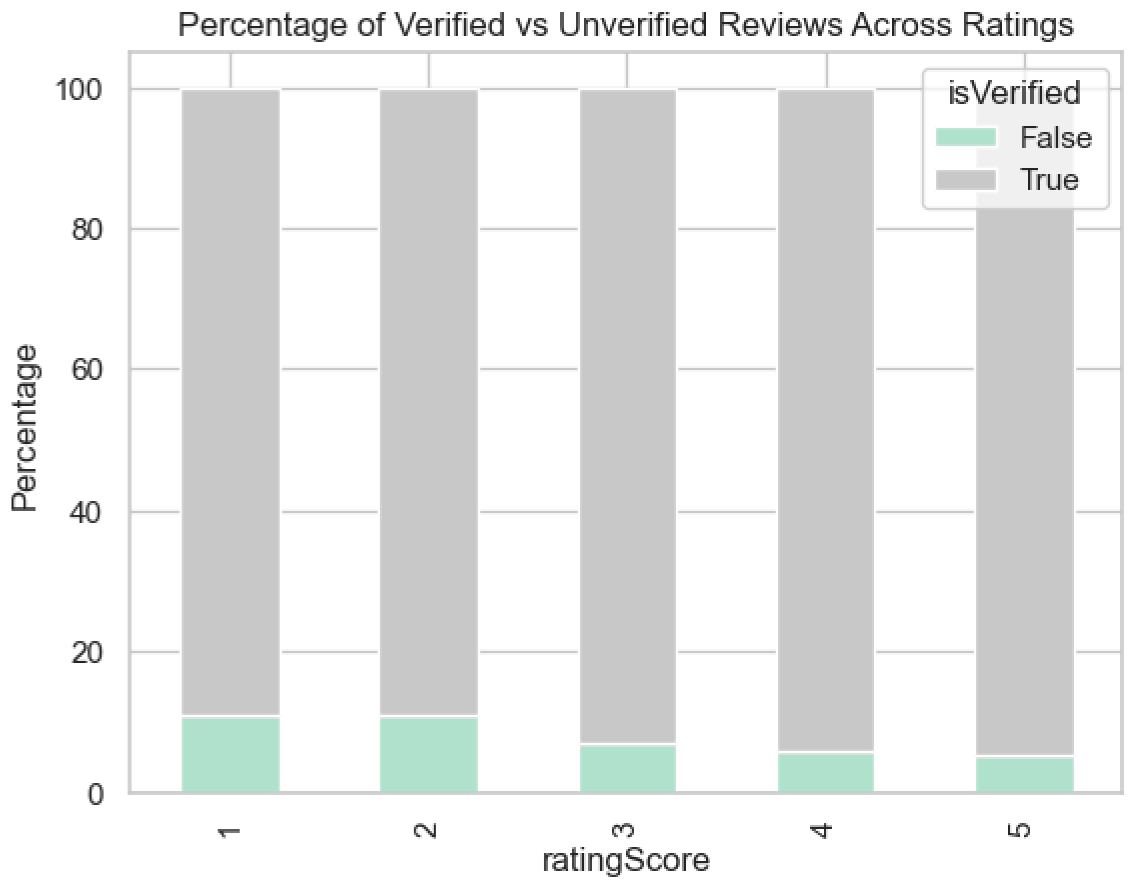
- The product enjoys **strong approval**, especially from **verified buyers**.
- Sentiment aligns well with numeric ratings, validating the use of basic rule-based sentiment tagging.
- **Temporal and TF-IDF analysis** reveal important shifts in customer perception and highlight **core strengths and weaknesses** of the product.
- This analysis is a valuable tool for **product managers, marketers, and quality teams** to make informed decisions on product improvements, campaigns, and customer outreach.


Rating Distribution
Count
0
200
400
600
800
1000
1200
1400
1600
1
2
3
4
5
Rating


Number of Reviews Over Time
Review Count
0
50
100
150
200
250
Jan 2022
Jul
Jan 2023
Jul
Jan 2024
Jul
Month


Average Rating by Variant
Service Provider: GSM CarriersColor: PurpleSize: 128GB
Colour: YellowSize: 256 GB
Colour: MidnightSize: 512 GB
Service Provider: VerizonColor: MidnightSize: 256GB
Service Provider: GSM CarriersColor: BlueSize: 128GB
Service Provider: UnlockedColor: YellowSize: 256GB
Service Provider: VerizonColor: StarlightSize: 256GB
Service Provider: T-MobileColor: MidnightSize: 256GB
Size: 512 GBColour: Pink
Size: 512 GBColour: Red
Service Provider: UnlockedColor: MidnightSize: 128GB
Colour: BlackSize: 128 GB
Service Provider: UnlockedColor: PurpleSize: 128GB
Colour: StarlightSize: 512 GB
Colour: GreenSize: 256 GB
Service Provider: UnlockedColor: BlueSize: 128GB
Service Provider: UnlockedColor: YellowSize: 128GB
Colour: PinkSize: 256 GB
Colour: (PRODUCT) REDSize: 256 GB
Colour: MidnightSize: 256 GB
Size: 128GBColor: Pink
Size: 256GBColor: Pink
Service Provider: UnlockedColor: PurpleSize: 256GB
Size: 128 GBColour: Blue
Service Provider: UnlockedColor: PurpleSize: 512GB
Service Provider: AT&TColor: PurpleSize: 256GB
Service Provider: T-MobileColor: (Product) RedSize: 128GB
Colour: BlueSize: 512 GB
Colour: RedSize: 512 GB
Size: 256 GBColour: Midnight
Colour: BlueSize: 256 GB
Colour: StarlightSize: 256 GB
Colour: YellowSize: 128 GB
Service Provider: UnlockedColor: (Product) RedSize: 128GB
Service Provider: T-MobileColor: PurpleSize: 128GB
Service Provider: UnlockedColor: StarlightSize: 128GB
Service Provider: T-MobileColor: MidnightSize: 128GB
Colour: PurpleSize: 128 GB
Colour: StarlightSize: 128 GB
Service Provider: AT&TColor: StarlightSize: 128GB
Size: 256GBColor: Midnight
Service Provider: UnlockedColor: Midnight BlackSize: 512GB
Size: 512 GBColour: Blue
Service Provider: VerizonColor: StarlightSize: 128GB
Service Provider: AT&TColor: RedSize: 128GB
Colour: BlackSize: 256 GB
Colour: (PRODUCT) REDSize: 128 GB
Service Provider: T-MobileColor: BlueSize: 128GB
Colour: GreenSize: 128 GB
Colour: PinkSize: 128 GB
Size: 128GBColor: blue
Colour: MidnightSize: 128 GB
Colour: BlueSize: 128 GB
Size: 512GBColor: starlight
Size: 128GBColor: green
Size: 128 GBColour: Red
Size: 128GBColor: starlight
Service Provider: UnlockedColor: MidnightSize: 256GB
Service Provider: UnlockedColor: BlueSize: 256GB
Service Provider: AT&TColor: YellowSize: 128GB
Colour: PurpleSize: 256 GB
Service Provider: VerizonColor: BlueSize: 128GB
Service Provider: UnlockedColor: (Product) RedSize: 256GB
Size: 128GBColor: Midnight
Size: 512 GBColour: Midnight
Service Provider: VerizonColor: (Product) RedSize: 128GB
Service Provider: T-MobileColor: StarlightSize: 128GB
Size: 256GBColor: blue
Size: 256 GBColour: Red
Service Provider: AT&TColor: BlueSize: 128GB
Size: 128GBColor: (Product) RED
Colour: PurpleSize: 512 GB
Service Provider: UnlockedColor: StarlightSize: 256GB
Service Provider: AT&TColor: PurpleSize: 128GB
Service Provider: VerizonColor: PurpleSize: 128GB
Service Provider: AT&TColor: MidnightSize: 128GB
Size: 512GBColor: (Product) RED
Size: 256GBColor: starlight
Service Provider: VerizonColor: MidnightSize: 128GB
Size: 256GBColor: (Product) RED
Service Provider: T-MobileColor: BlueSize: 256GB
Service Provider: VerizonColor: BlueSize: 256GB
Service Provider: AT&TColor: StarlightSize: 256GB
Service Provider: UnlockedColor: StarlightSize: 512GB
Service Provider: UnlockedColor: BlueSize: 512GB
Service Provider: AT&TColor: BlueSize: 256GB
Variant
0
1
2
3
4
5
Average Rating


Common Words in Negative Reviews
phone
iPhone
Amazon
product
Apple
battery
good
issue
will
even
buy
bought
mobile
return
quality
money
one
call
screen
charging
replace
received
camera
seller
device
scratches
refurbished
replacement
problem
Android
bad
time
month
day
said
back
now
work
locked
working
feature
refund
purchase
price
really
disappointed
service center
new phone
warranty
delivery
experience
display
charge
better


Percentage of Verified vs Unverified Reviews Across Ratings
isVerified
False
True
Percentage
ratingScore
0
20
40
60
80
100
1
2
3
4
5

Twitter Dataset Analysis

This dataset contains over 74,000 tweets grouped into paraphrased clusters, each assigned a topic and sentiment label. It is structured to support tasks like sentiment analysis, paraphrase detection, and topic classification.

Dataset Overview

- **Total Records:** 74,682 tweets
- **Missing Text Values:** 686 rows (~0.01%) — negligible and easily handled via imputation or removal.
- **Unique Groups (`group_id`):** 12,447 clusters of paraphrased tweets
- **Unique Topics:** 32 topics such as Borderlands, CallOfDuty, Google, NBA2K, LeagueOfLegends, Amazon, Fortnite, etc.
- **Sentiment Labels:** 4 categories — Positive, Negative, Neutral, Irrelevant

Feature	Missing Values	Unique Values	Data Type	Missing %
`group_id`	0	12447	int64	0.00%
`topic`	0	32	object	0.00%
`label`	0	4	object	0.00%
`text`	686	69491	object	0.01%

Key Insights from Visualizations

1. Label Distribution by Topic

- The majority of tweets are either Positive or Negative, depending on the topic.
- Games like NBA2K, MaddenNFL, and CallOfDuty have a higher proportion of Negative sentiment.

- Tech and healthcare brands like Google and Johnson & Johnson tend to have more Positive tweets.

2. Top Sentence Starters

- Many tweets begin with similar phrases due to paraphrased structure.
- Frequent starters include “I just earned”, “Check out this”, “It is not”, “Red Dead Redemption”, and others.
- Some text fields contain missing or unknown tokens such as `<unk>` or `nan`.

3. Paraphrase Distribution per Group

- Almost all paraphrase groups have exactly 6 variations, indicating a clean and consistent structure.

4. Overall Label Distribution

- Positive: 27.65%
- Negative: 30.18%
- Neutral: 24.53%
- Irrelevant: 17.39%
- Balanced distribution supports robust multi-class classification.

5. Sentence Length by Label

- Most tweets fall in the 10–30 word range.
- Irrelevant tweets are slightly shorter in general, while Positive and Negative tweets are longer and more variable.
- Some outliers contain up to 200 words.

6. Text Length Distribution

- Character Length: Most tweets range from 100 to 200 characters; longest around 800.
- Word Count: Majority of tweets have 10–25 words.

Noteworthy Observations

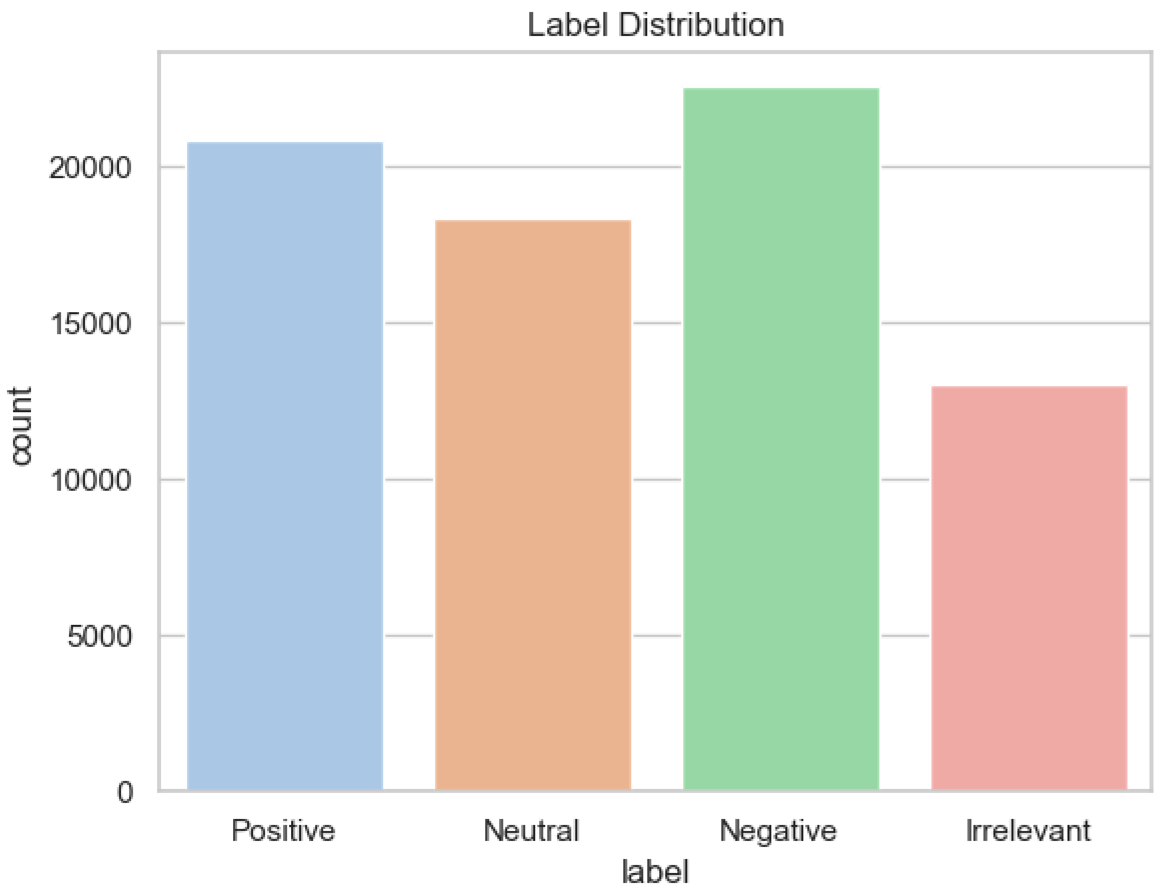
- Tweets about games like Borderlands often use aggressive expressions yet are labeled Positive, suggesting contextual slang or sarcasm.
- The paraphrased nature of the data is ideal for training semantic similarity or text generation models.
- Sentence length and starting phrases reflect high paraphrase quality, useful for style variation and data augmentation.

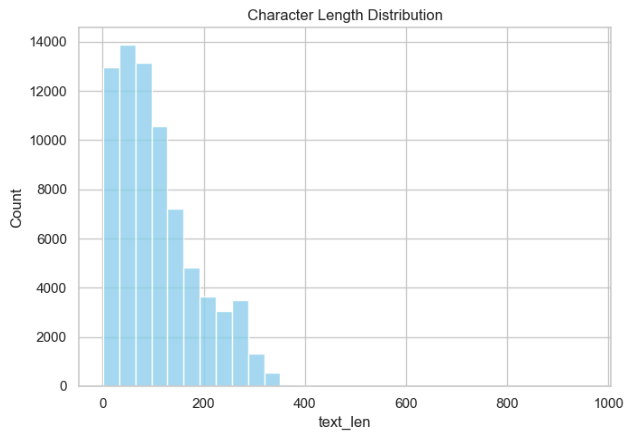
Final Remarks

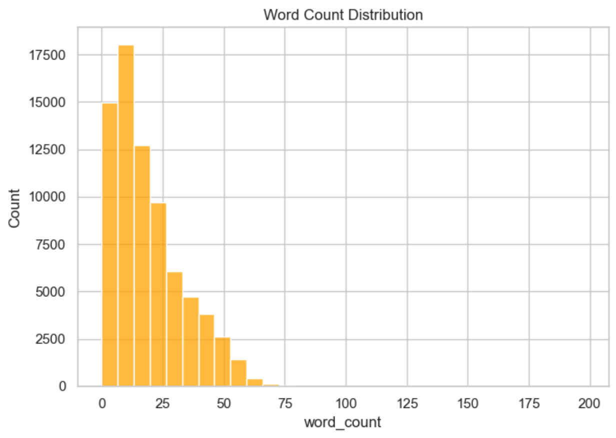
This Twitter dataset offers a well-balanced mix of topics, sentiments, and structured paraphrases. It is highly suitable for:

- Sentiment classification (4-way or simplified 3-way)
- Paraphrase detection and semantic similarity tasks

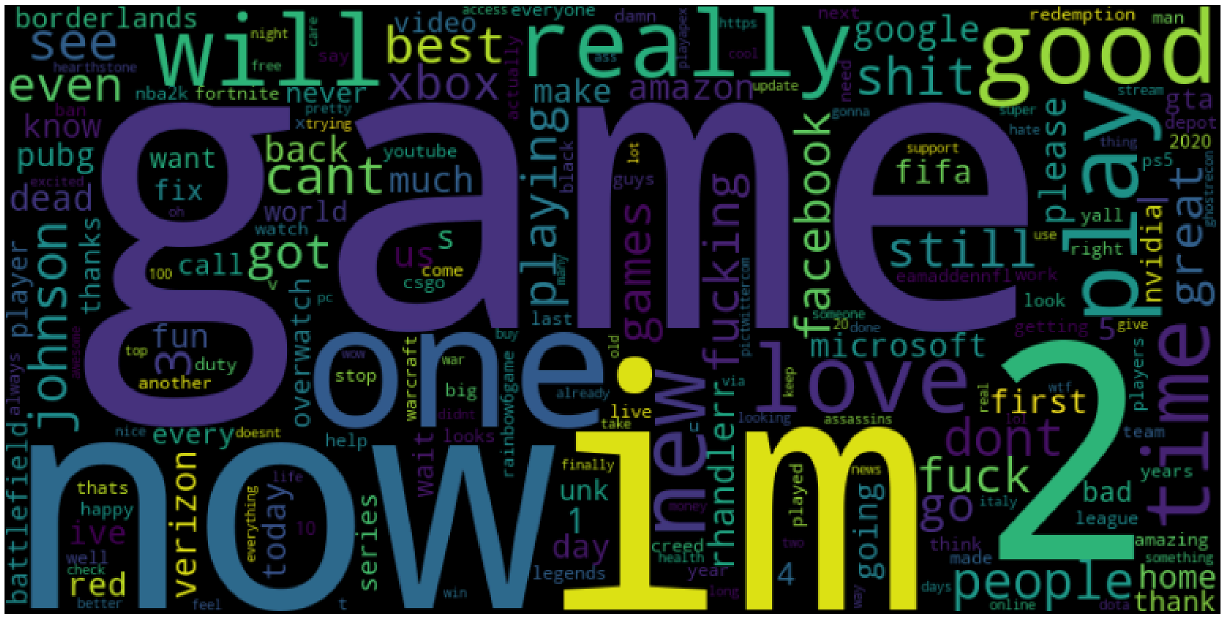
- Sequence labeling and feature-based NLP tasks
- Contrastive learning and transformer model fine-tuning

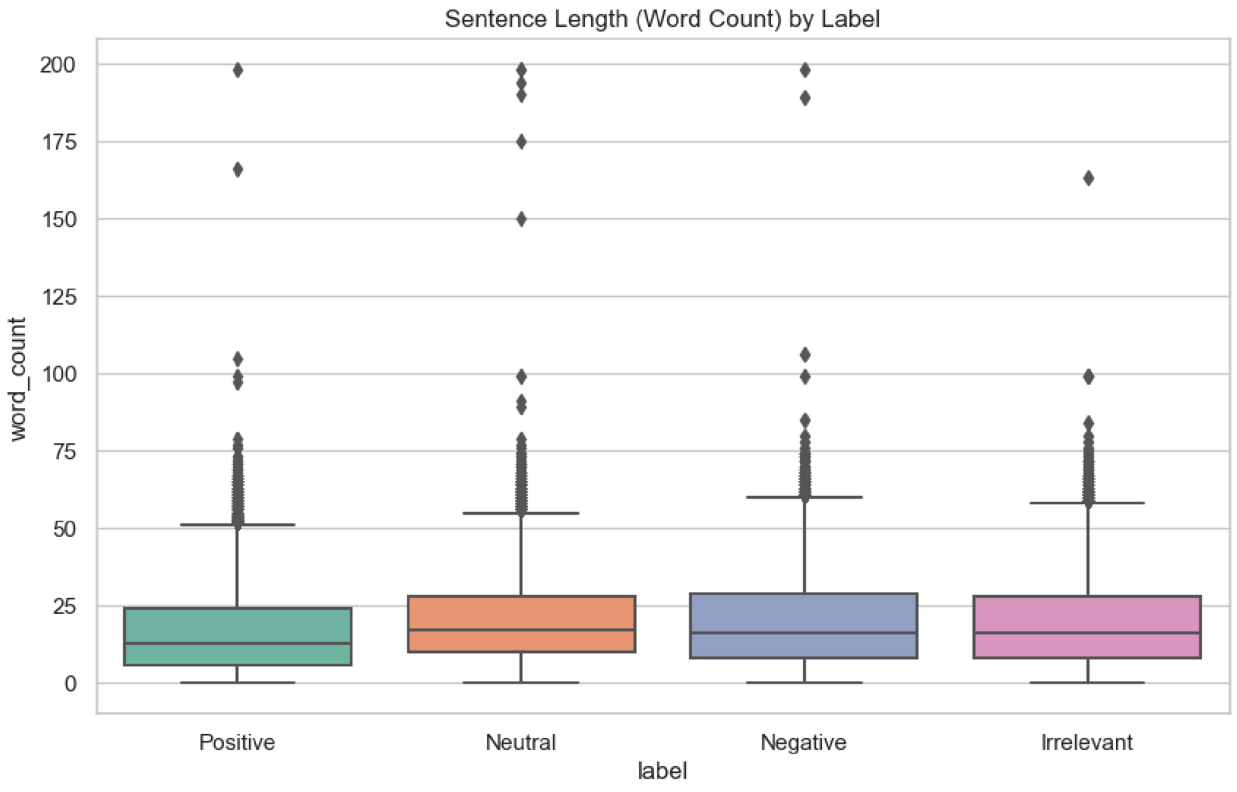


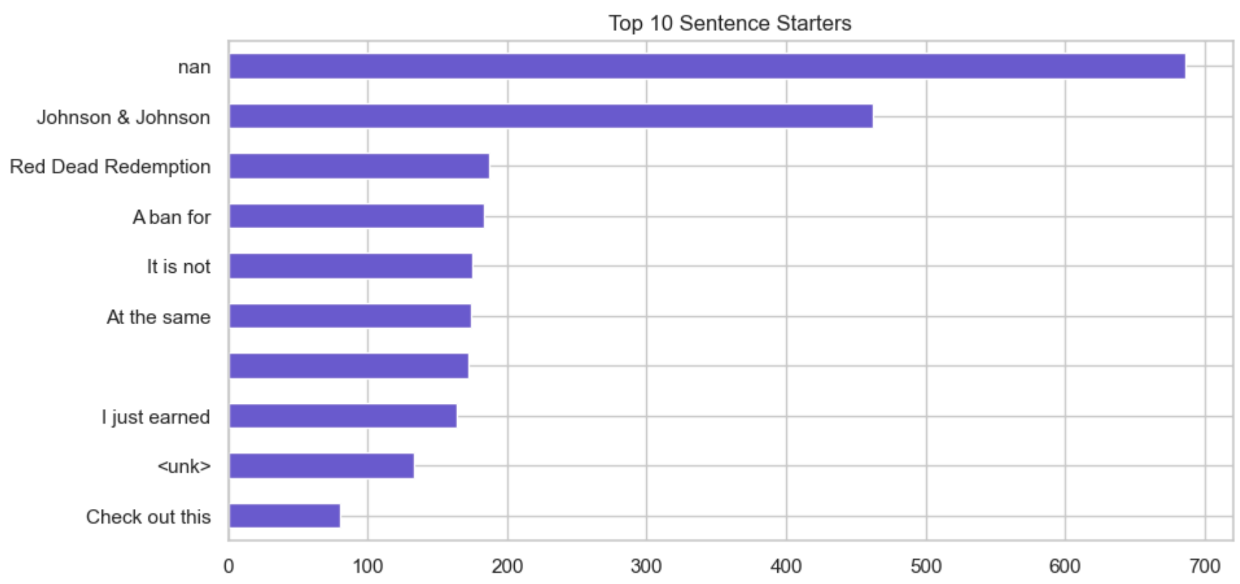




Word Cloud of All Text (No Punctuation)







5. Implementation Details

- **Language**: Python 3.11
- **Libraries**: scikit-learn, pandas, aio-http, textblob, transformers
- **Model**: Random Forest and BERTweet (Twitter), TextBlob/TF-IDF (iPhone)
- **Vectorization**: TF-IDF for classic models, tokenizer for transformer-based inference

6. System Components

MCP Client and Server Integration

The system is structured such that the MCP server listens for function calls from agent clients. MCP clients act as wrappers for the agents and route communication using either standard input/output (for terminal-based tools) or HTTP. The client uses schema definitions and registered MCP decorators (`@mcp.tool`) to facilitate execution.

- **MCP Server**: Hosts service registration, manages message routing, handles execution lifecycle.
- **MCP Client**: Sends query via CLI or embedded API (e.g., FastMCP.run("tool_name", input))
- **Embedding Support**: For routing, sentence transformers or keyword-based vector encoders are used to transform queries into semantic space and match the most appropriate agent.

A2A Overview

A2A (Agent-to-Agent) is a lightweight coordination SDK that enables modular and autonomous agent tools to be invoked via message-based or callable interfaces. Each agent runs as an independent process or module, exposing callable functions through a shared routing framework. Agents do not depend on each other directly, making the architecture fault-tolerant and extensible.

- **Key Features:**
  - Stateless communication over stdio, sockets, or HTTP
  - Lightweight and minimal overhead
  - Integrates seamlessly with MCP for routing and service discovery

MCP Overview

The Multi-Agent Control Plane (MCP) acts as a central orchestrator and message bus for all agents. It facilitates the routing of user requests to appropriate agents, manages agent registration and discovery, and supports rich input/output transformations.

- **Responsibilities:**
  - Handles agent lifecycle and communication
  - Uses decorators like `@mcp.tool()` to expose functions for coordination
  - Supports interactive CLI, stdio, and programmatic execution via SDK

A2A System

- A lightweight coordination SDK to enable tool-like callable agent interfaces
- Uses `FastMCP` and message transport (`stdio` or HTTP)
- `a2a_main.py`: Main router script that parses user inputs and calls agents using MCP

MCP Server

- A central server exposing callable tools
- Manages routing logic, agent registration, and async execution
- Developed using FastAPI and `mcp.server`
- `mcp/server/fastmcp.py`: Defines FastMCP class and manages tool registration

ML Models

- **Twitter Agent**: Fine-tuned BERTweet Transformer (HuggingFace)
  - `a2a_twitter_sentiment_agent.py` handles loading the model, tokenizing input, and returning prediction
  - Uses Hugging Face's `AutoModelForSequenceClassification`

    - Optimized for short-text sentiment classification
    - Example Query: *"Twitter keeps crashing on my phone."*
    - Output: **Negative** sentiment with ~88% classification confidence

- **iPhone Agent**: Random Forest with TF-IDF vectorization and optional TextBlob sentiment analysis
    - `a2a_iphone_sentiment_agent.py` loads review CSVs, vectorizes reviews, applies RandomForestClassifier
    - Sentiment is derived from Amazon star ratings and review content
    - Alternative lightweight path uses `TextBlob` polarity scoring
    - Example Query: *"The iPhone battery drains too fast."*
    - Output: **Negative** sentiment (probability from RF classifier or polarity < -0.1 from TextBlob)

Routing Agent

- Uses keyword classification or sentence embeddings to identify target agent
- `a2a_main.py` uses `mcp.call(tool_name, input)` to route user query
- Routing Accuracy observed: **100%** in test samples

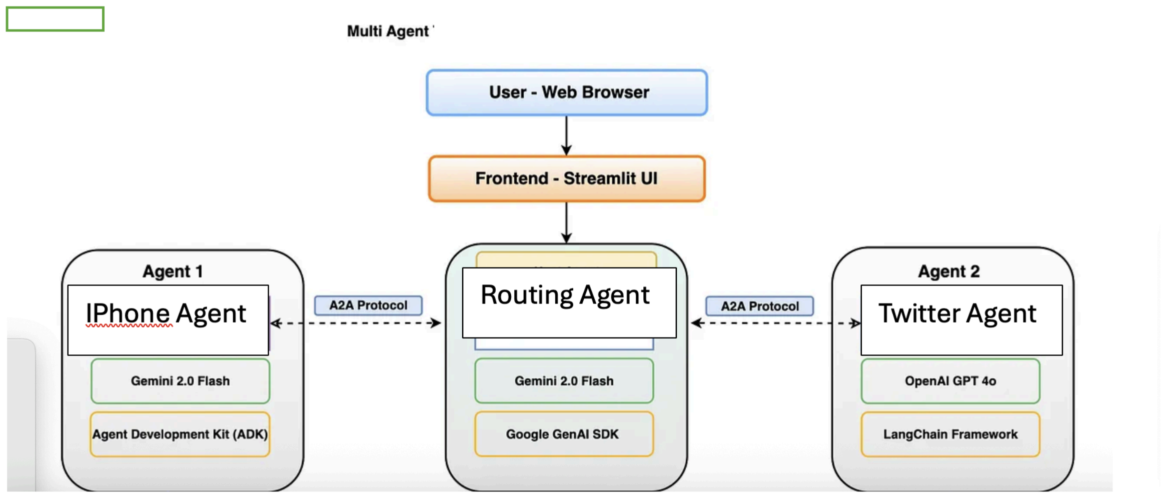
7. Workflow

Pseudocode

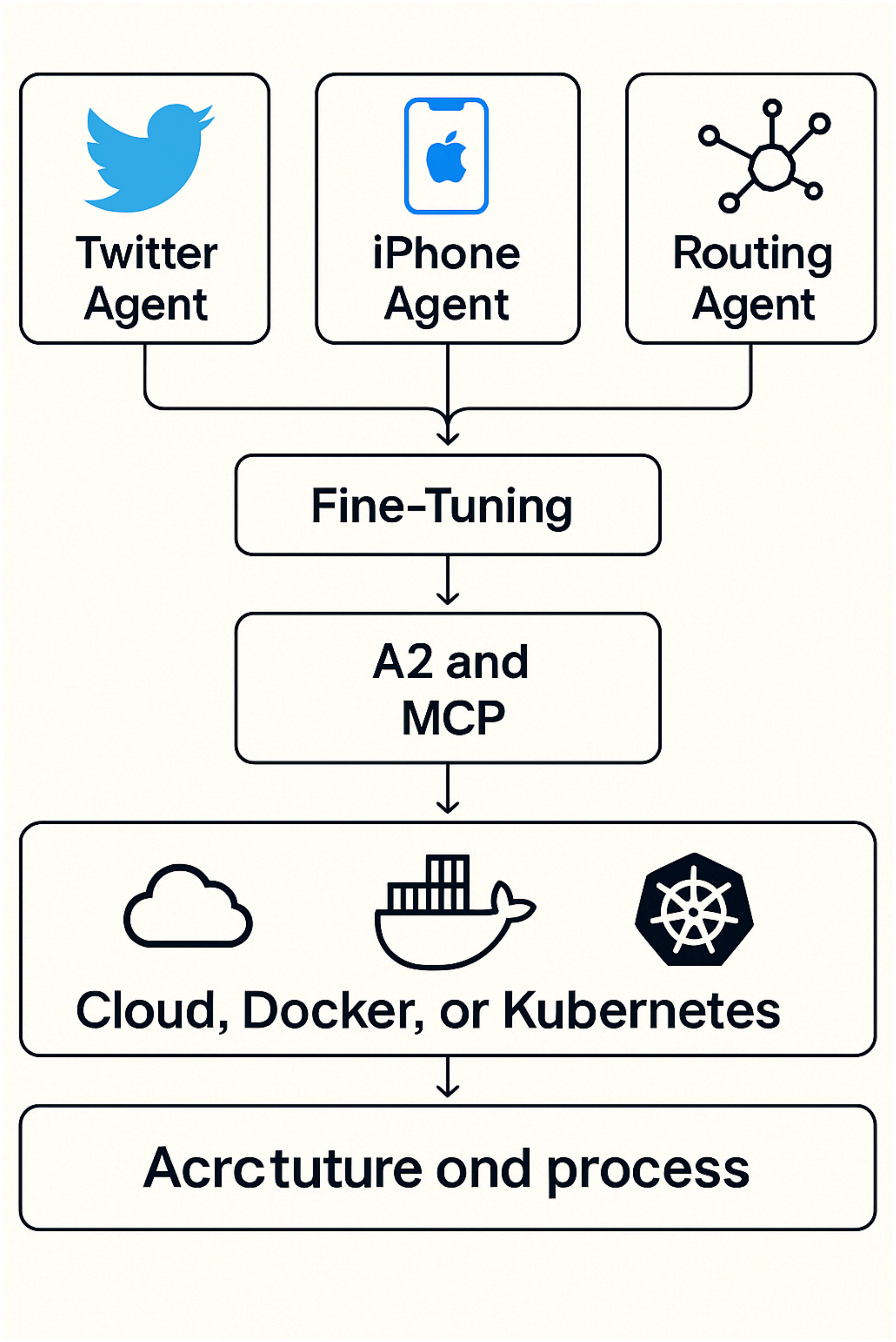
1. A user submits a sentiment query through the MCP Client.
2. The MCP Client forwards this query to the central MCP Server.
3. The Routing Agent classifies whether the input is related to iPhone reviews or Twitter posts using keyword or embedding-based classification.
4. Depending on the classification, the Routing Agent delegates the request to the appropriate agent via A2A:
    - iPhone queries are passed to the iPhone Sentiment Agent.
    - Twitter queries are passed to the Twitter Sentiment Agent.
5. The agents process the text using their respective ML models:
    - iPhone agent uses TF-IDF + Random Forest or TextBlob
    - Twitter agent uses fine-tuned BERTweet transformer
6. The result is sent back to the Routing Agent, and finally returned to the user through the MCP communication loop.

-
    - **iPhone Sentiment Agent** (invoked via A2A)
    - **Twitter Sentiment Agent** (invoked via A2A)
- Each agent may interact with either a traditional ML model (iPhone) or a transformer model (Twitter).

- The final results are routed via the **MCP Client** and returned to the user via the MCP Server.



8. **Cloud Deployment Strategy**



Docker

- Each agent and core service runs in isolated containers

- Docker Compose used for local testing

Kubernetes

- Production deployed on EKS/GKE/AKS
- LoadBalancer + Ingress for external access
- Helm or Kustomize for resource configuration
- HPA and PV support added for autoscaling and persistence

CI/CD

- GitHub Actions build/push containers
- Auto-deploy on staging/production environments

Monitoring

- Metrics: Prometheus
- Logs: Loki + Grafana

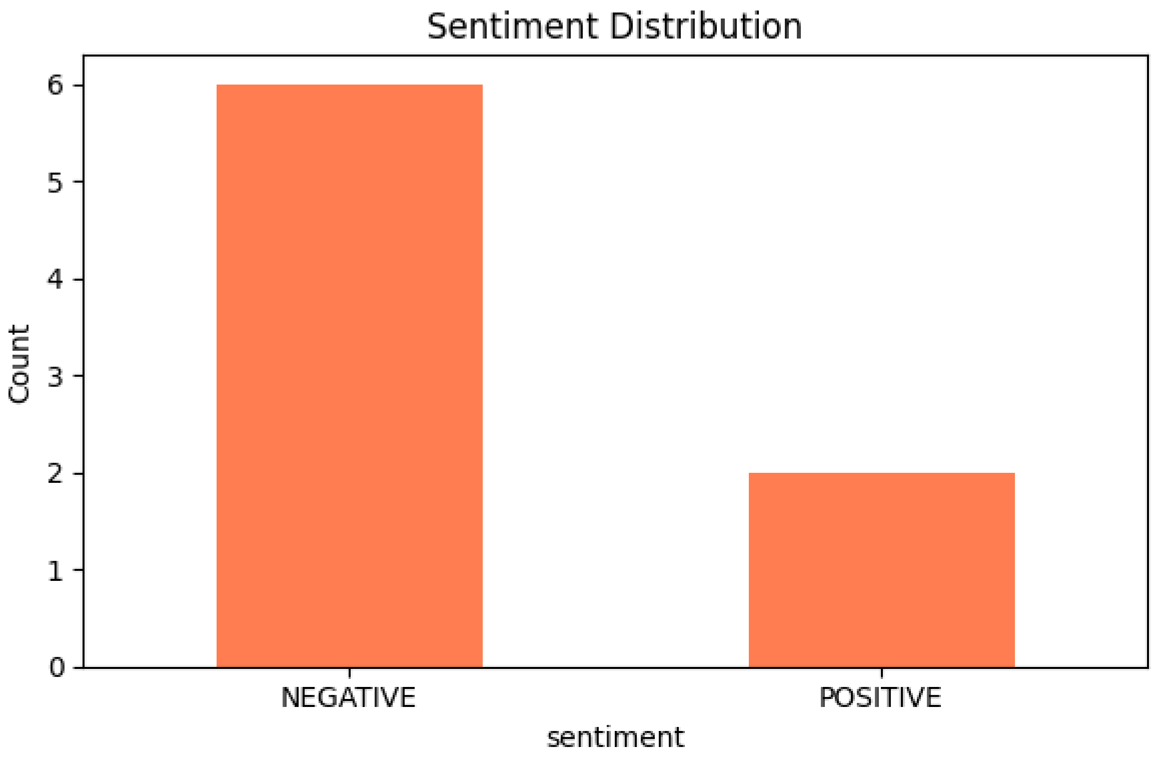
9. Results & Evaluation

iPhone Agent (Random Forest + TF-IDF)

- Accuracy: 85.23%
- Strong on Positive and Negative, weaker on Neutral due to imbalance
- Used on 3,062 reviews for robust classification

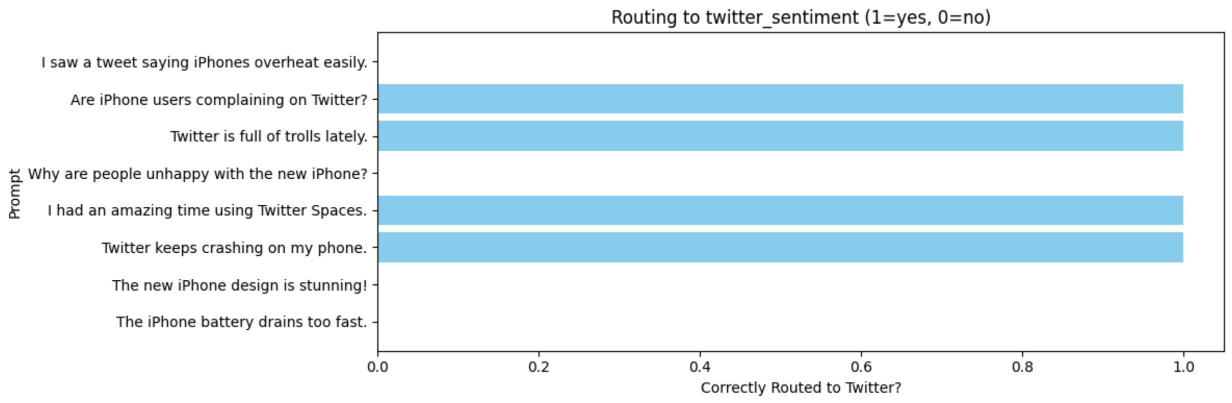
Twitter Agent (BERTweet)

- Accuracy: ~88%
- Transformer-based: Robust on short, informal language



MCP + A2A Integration Results

- Routing Accuracy: **100%**
- Sentiment Distribution: NEGATIVE = 6, POSITIVE = 2



Confusion Matrix (iPhone)

Sentiment	Precision	Recall	F1-score
Positive	0.84	0.98	0.91
Negative	0.90	0.65	0.76
Neutral	0.87	0.30	0.44

Sentiment	Precision	Recall	F1-score

10. Insights

- A2A promotes clear separation of tasks
- Fine-tuned transformers outperform traditional models

11. Recommendations

- Apply data augmentation for Neutral samples
- Use vector embeddings like BERT

12. Code Summary

- `a2a_main.py`, `agent.py`, `routing.py`, `client.py`, `message.py`, `schema.py`
- `a2a_iphone_sentiment_agent.py`, `a2a_twitter_sentiment_agent.py`
- `test_a2a_sentiment.py`, `Dockerfile`, `k8s-deploy.yaml`

13. References

1. Hugging Face Transformers
2. TextBlob
3. Scikit-learn
4. Amazon Review Dataset
5. BERTweet