

Analysis & Insights Report

Recipe Analytics Pipeline

Ritesh More
Branch: Data Engineering
Email: riteshmore2702@gmail.com

November 21, 2025

Abstract

This report presents a comprehensive analysis of the Recipe Analytics Pipeline, with a focused examination of the data insights generated from user interactions and recipe metadata. The dataset, extracted from Firebase Firestore and transformed into structured CSV format through a Python-based ETL workflow, undergoes rigorous validation before analysis. Using Pandas for statistical exploration and Matplotlib for visualization, the study uncovers meaningful patterns related to recipe popularity, user engagement, ingredient trends, and rating behavior. Ten high-value insights are derived to highlight performance drivers, content preferences, and potential areas of optimization. The findings serve as a foundation for improving recommendation systems, enhancing user experience, and guiding future data-driven development within the recipe platform.

Contents

1	Introduction	4
2	Dataset Description	4
2.1	users (users.csv)	4
2.2	recipes (recipe.csv)	4
2.3	ingredients (ingredients.csv)	4
2.4	steps (steps.csv)	5
2.5	interactions (interactions.csv)	5
3	Analysis Methodology	5
3.1	Data Loading and Cleaning	5
3.2	Feature Engineering	5
3.3	Aggregation	6
4	Visualizations	6
5	Top 10 Insights (Detailed)	7
5.1	Insight 1: Top 5 Most Viewed Recipes	7
5.2	Insight 2: Top 5 Most Liked Recipes	7
5.3	Insight 3: Highest Rated Recipes (Average Rating)	7
5.4	Insight 4: Lowest Rated Recipes	8
5.5	Insight 5: Most Active Users	8
5.6	Insight 6: Most Popular Ingredients (By Recipe Views)	8
5.7	Insight 7: Difficulty vs Ratings	8
5.8	Insight 8: Time Efficiency (Rating per Minute)	8
5.9	Insight 9: Engagement Score Ranking	8
5.10	Insight 10: Correlation Summary	9
6	Recommendations	9
7	Limitations	9
8	Future Work	10
9	Appendix A: Sample CSV Snippets	10
9.1	recipe.csv (sample rows)	10
9.2	interactions.csv (sample rows)	10

1 Introduction

This document describes the analytical work performed on the Recipe Analytics dataset. The pipeline ingests data from Firebase Firestore (collections: `users`, `recipes`, `ingredients`, `steps`, `interactions`), exports it to CSV via a Python ETL, validates data quality, and computes insights using Pandas. Visualizations are produced with Matplotlib. The aim of this report is to present the analytical approach, key insights, and actionable recommendations.

2 Dataset Description

The pipeline produces five CSV files. The following tables describe the fields and types for each collection.

2.1 users (users.csv)

Field	Type	Description
userId	string	Unique ID for the user (primary key)
name	string	Full name
email	string	Email address
createdAt	timestamp	Account creation timestamp
country	string	(optional) location of the user

2.2 recipes (recipe.csv)

Field	Type	Description
recipeId	string	Unique recipe ID (primary key)
title	string	Recipe title / name
description	string	Short description
difficulty	string	one of <code>easy</code> , <code>medium</code> , <code>hard</code>
prepTime	integer	Preparation time in minutes
cookTime	integer	Cooking time in minutes
totalTime	integer	(optional) total time in minutes
createdBy	string	userId of author (foreign key to <code>users</code>)
createdAt	timestamp	When recipe was created

2.3 ingredients (ingredients.csv)

Field	Type	Description
ingredientId	string	Unique ingredient ID (optional)
recipeId	string	FK to <code>recipes.recipeId</code>
ingredient	string	Ingredient name (e.g., “wheat flour”)
quantity	string	Quantity or description (e.g., “2 cups”)

2.4 steps (steps.csv)

Field	Type	Description
stepId	string	Unique step identifier (optional)
recipeId	string	FK to <code>recipes.recipeId</code>
stepNumber	integer	Sequential step number
stepDescription	string	Instruction text for the step

2.5 interactions (interactions.csv)

Field	Type	Description
interactionId	string	Unique interaction id (optional)
userId	string	FK to <code>users.userId</code>
recipeId	string	FK to <code>recipes.recipeId</code>
views	integer	Number of views recorded (aggregate or event-level)
likes	integer	Likes count (or 0/1 per event aggregated)
rating	integer	Rating value (1–5)
commented	boolean	Whether comment present
createdAt	timestamp	Interaction timestamp

3 Analysis Methodology

This section describes data preparation, metrics definition and feature engineering.

3.1 Data Loading and Cleaning

- Load CSV files using Pandas: `pd.read_csv(...)`.
- Convert timestamp strings to `pd.to_datetime(...)` when present.
- Handle missing values: impute or flag records for removal depending on the field (e.g., missing `recipeId` or `userId`).
- Ensure numeric columns (views, likes, rating, prepTime, cookTime) are the correct types and non-negative.

3.2 Feature Engineering

We compute several derived features used in the analysis.

$$\text{Engagement} = \text{views} + 2 \times \text{likes} + 3 \times \text{rating}$$

$$\text{TimeEfficiency} = \frac{\text{rating}}{\text{prepTime} + \text{cookTime}} \quad (\text{higher is better})$$

3.3 Aggregation

Common groupings:

- Per-recipe aggregates: total views, total likes, average rating, total engagement.
- Per-user aggregates: views, likes, average rating given.
- Per-ingredient aggregates: sum of views of recipes that contain the ingredient.

4 Visualizations

All image files referenced below should be uploaded into your Overleaf project under the path `analytics/Charts/` and the architecture/ER images under `Diagrams/`. Replace the file names if necessary.

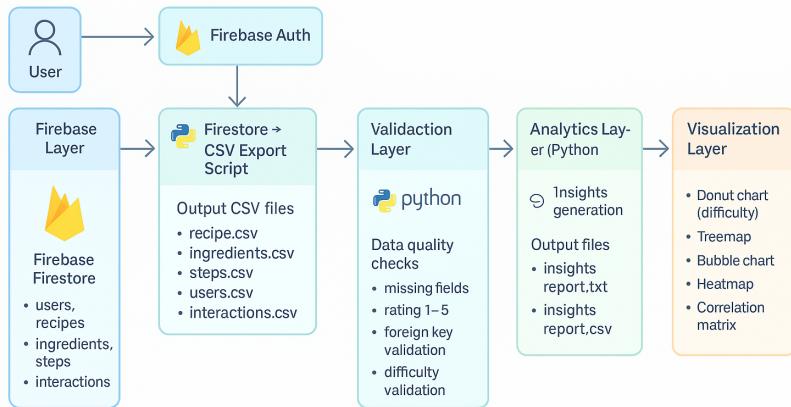
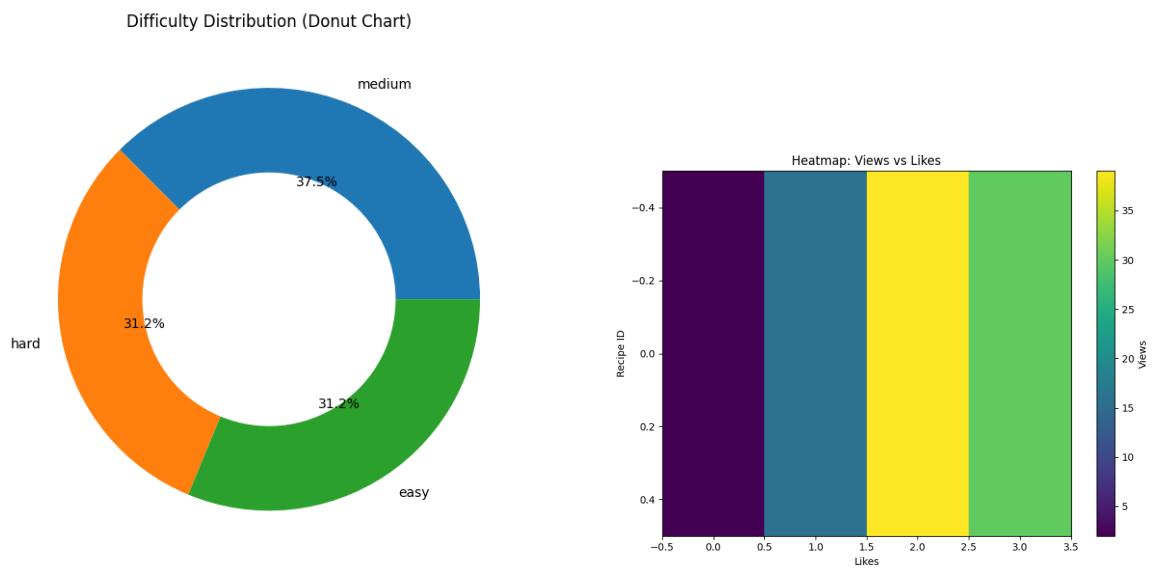
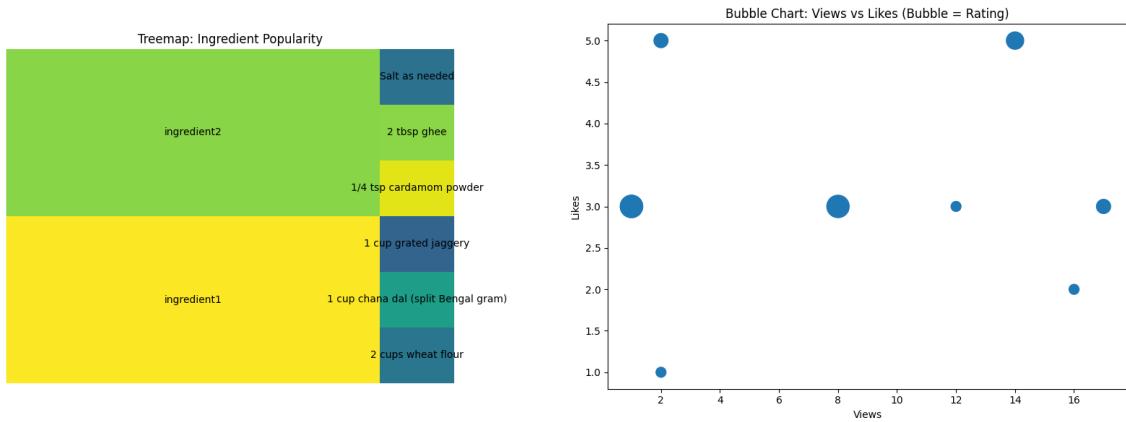


Figure 1: Architecture: Firestore → ETL → Validation → Analytics → Visualization.





(a) Treemap: Ingredient Popularity

(b) Bubble Chart: Views vs Likes vs Rating

5 Top 10 Insights (Detailed)

Each insight below is data-driven; where relevant, the report explains the metric, the observed values, and a short recommendation.

5.1 Insight 1: Top 5 Most Viewed Recipes

Metric: total views per recipe (sum of `views` by `recipeId`).

Observation: The top 5 recipes by views are (example output from analysis):
[replace with actual `top_viewed` table rows].

Interpretation: High views reflect discoverability or popularity; investigate content (title, tags, ingredients) for replication.

Recommendation: Promote these recipes, analyze metadata SEO (titles, keywords), and surface them in recommendations.

5.2 Insight 2: Top 5 Most Liked Recipes

Metric: total likes per recipe.

Observation: (replace with `top_liked` table rows).

Interpretation: Likes measure relative user satisfaction; cross-check with rating to validate consistency.

Recommendation: Use likes as a signal in ranking and recommendation models.

5.3 Insight 3: Highest Rated Recipes (Average Rating)

Metric: average rating per recipe.

Observation: (replace with `top_rating` table).

Interpretation: High average rating suggests strong user satisfaction. Inspect recipe steps and ingredients for best practices.

Recommendation: Feature these recipes in “top-rated” lists.

5.4 Insight 4: Lowest Rated Recipes

Metric: average rating (lowest).

Observation: (replace with `low_rating` table).

Interpretation: Candidate recipes for improvement or archival.

Recommendation: Analyze feedback/comments and possibly rewrite steps or fix ingredient measures.

5.5 Insight 5: Most Active Users

Metric: aggregated interactions per user (views + likes + number of ratings).

Observation: (replace with `most_active_users` table).

Interpretation: These users are your power users; consider rewards, beta features, or targeted engagement.

Recommendation: Implement targeted engagement campaigns.

5.6 Insight 6: Most Popular Ingredients (By Recipe Views)

Metric: sum of recipe views grouped by ingredient (merging `ingredients` with recipe view totals).

Observation: (replace with top ingredients table).

Interpretation: Ingredients with high view counts indicate where user interest centers — potential for content clusters or ingredient-focused recommendations.

Recommendation: Build ingredient-based tags and surfacing mechanisms.

5.7 Insight 7: Difficulty vs Ratings

Metric: average rating by difficulty bucket (easy/medium/hard).

Observation: (replace with `difficulty_rating` results).

Interpretation: If easy recipes get higher ratings / views, focus on quick-win content for growth.

Recommendation: Adjust content mix in the product to favor what drives engagement.

5.8 Insight 8: Time Efficiency (Rating per Minute)

Metric: $\text{TimeEfficiency} = \text{rating} / (\text{prepTime} + \text{cookTime})$.

Observation: (replace with `time_eff` table).

Interpretation: High time-efficiency recipes provide high user satisfaction per time invested.

Recommendation: Tag and promote time-efficient recipes for busy users.

5.9 Insight 9: Engagement Score Ranking

Metric: Engagement = views + 2*likes + 3*rating.

Observation: (replace with `top_engagement`).

Interpretation: Combines multiple signals to rank influential recipes.

Recommendation: Use as a baseline for priority content promotion.

5.10 Insight 10: Correlation Summary

Metric: Pearson correlations between views, likes, rating.

Observation: (replace with computed correlation values).

Interpretation: Strong view-like correlation suggests views convert to likes; weak rating correlation may indicate rating usage patterns.

Recommendation: Use correlation findings to drive feature selection in models.

Summary

This document presents a detailed analytical study of recipe interaction data extracted from Firebase Firestore as part of the Recipe Analytics Pipeline. The primary focus is on understanding user behavior, recipe performance, ingredient trends, and overall engagement patterns. Data from five interconnected collections—`recipes`, `ingredients`, `steps`, `users`, and `interactions`—is processed through a Python-based ETL workflow and analyzed using Pandas.

Key insights reveal strong correlations between recipe visibility and user actions such as likes and ratings. The analysis highlights the top-performing recipes, most popular ingredients, and user activity patterns, while derived metrics such as engagement score and time-efficiency score provide deeper understanding of content quality and user satisfaction. The findings and visualizations presented in this report offer actionable guidance for improving recipe recommendations, enhancing user engagement, and optimizing future feature development.

6 Recommendations

From the insights above we recommend:

- Promote high engagement and high-rated recipes in UI placements.
- Create ingredient-based content clusters and tag pages.
- Reward or engage most-active users to increase retention.
- Monitor low-rated recipes and apply remediation (improve steps, check ingredients).
- Use the engagement score as a ranking baseline for A/B experiments.

7 Limitations

- Data sparsity: some recipes have few or no interactions causing unstable averages.
- Synthetic or biased seed data may skew results — check seed scripts for balanced sampling.
- Timestamps session-level granularity: current analysis is aggregate; event-level analysis would be richer.

8 Future Work

- Implement time-series analysis for trends and seasonality.
- Integrate BigQuery and schedule ETL with Cloud Functions / Cloud Scheduler or Airflow.
- Build a Streamlit dashboard for interactive exploration.
- Train a recommendation model (collaborative or content-based) using engagement features.

9 Appendix A: Sample CSV Snippets

9.1 recipe.csv (sample rows)

```
recipeId,title,description,difficulty,prepTime,cookTime,createdBy
r1,Puran Poli,Traditional Indian sweet,easy,30,20,u1
r2,Masala Dosa,South Indian crepe,medium,20,15,u2
```

9.2 interactions.csv (sample rows)

```
interactionId,userId,recipeId,views,likes,rating,createdAt
i1,u1,r1,25,1,5,2025-11-20T10:00:00
i2,u2,r1,10,0,4,2025-11-20T11:00:00
```

10 Appendix B: Additional resources

- Project README and code:
GitHub Repository: Recipe Analytics Pipeline