OPEN ACCESS INTERNATIONAL JOURNAL OF SCIENCE & ENGINEERING

SMART VIEW - AI -POWERED VISUALIZATION & REVIEWING & RECOMMENDATION OF MULTI APP

Pradip A. Patil[1],Pranav D. Lokhande[2],Ashwin D. Lokhande[3],Ankita N. Shete[4],Anishka T. Waghmare[5]

Prof, Computer Engineering,Sandip Institute Of Technology and Research Center Nashik(SITRC)[1]

Student, Computer Engineering,Sandip Institute Of Technology and Research Center Nashik(SITRC)[2 3 4 5]

pradippatil@gmail.com[1],pranavlokhande22@gmail.com[2],ashwindl2208@gmail.com[3],ankitashete109@gmail.com[4],

anishkawaghmare@gmail.com[5]

***Abstract:** In the modern mobile application ecosystem, user reviews play a critical role in shaping application quality, user retention, and overall market success. However, these reviews are typically unstructured, voluminous, and spread across multiple platforms, making them difficult for developers to analyze and act upon.This project proposes "AI-Powered Visualization & Recommendation of Multi-App Reviews", an offline system that leverages Natural Language Processing (NLP) and Machine Learning (ML) techniques to transform raw user reviews into meaningful insights. Reviews are ingested through local CSV/Excel files and stored in a lightweight SQLite database, ensuring complete independence from third-party APIs or external services. The system applies preprocessing steps such as tokenization, stopword removal, and lemmatization, followed by a sentiment classification pipeline using TF-IDF and supervised learning models (Logistic Regression, SVM). The processed results are presented through an interactive GUI-based dashboard that features pie charts, bar graphs, and temporal trend visualizations, enabling developers to track user sentiments over time and across versions. In addition, the system incorporates a recommendation engine that identifies recurring issues (e.g., crashes, UI problems, battery drain) and translates them into prioritized, actionable suggestions for developers. Export functionality further supports local reporting in CSV/PDF-formats.*

By unifying offline sentiment analysis, visualization, and recommendation, the proposed system empowers developers with data-driven decision-making while ensuring privacy, reliability, and ease of deployment in academic and industrial contexts.

***Keywords** Sentiment Analysis, Opinion Mining, Natural Language Processing, Machine Learning, SQLite, GUI, Data Visualization, App Reviews, Recommendation Engine, Offline Analytics*

I. INTRODUCTION

1.1 Background & Motivation

1. Mobile applications dominate today's digital economy, serving as platforms for communication, commerce, education, entertainment, and financial services.
2. With millions of apps available on Google Play Store, Apple App Store, and Microsoft Store, developers rely heavily on user reviews to gauge application quality, feature requests, and potential issues.
3. These reviews influence user retention, app ratings, and market competitiveness. However, they are often unstructured, noisy, and spread across platforms, making it challenging to extract actionable insights.
4. Traditional sentiment analysis systems exist, but most are either limited to binary classification (positive/negative) or depend on third-party APIs/cloud services, raising issues of cost, privacy, and dependence.
5. There is a need for an offline, lightweight, yet intelligent framework that can process reviews, visualize trends, and generate developer-friendly recommendations in a structured manner.

1.2 Problem Context

1. Developers face information overload with thousands of daily reviews that cannot be read manually.
2. Existing tools focus only on sentiment classification but fail to:

- Differentiate between app aspects (e.g., UI, performance, battery).
- Provide trend analysis across time and versions.

3. Convert findings into actionable recommendations for developers.

4. Many solutions depend on third-party APIs (Google/Apple APIs, cloud NLP services), which:

- Compromise privacy.
- Increase deployment cost.
- Violate academic project feasibility (internet dependency).

II.OBJECTIVE

1.Unified Local Data Hub

Design a clean SQLite schema (apps, reviews, sentiments, suggestions) and achieve ≤2-sec import for a 10k-row CSV with basic validation (missing fields, date format).

2.Robust NLP Cleaning Pipeline

Implement an offline text pipeline (lowercasing, punctuation/URL removal, stopwords, lemmatization) with a toggleable config, targeting ≥95% successful parses and a <300ms average clean time per 1k reviews (batched).

3.Baseline Sentiment Classifier (Benchmark-Ready)

Train TF-IDF + Logistic Regression/SVM and report Accuracy + Macro-F1 on a hold-out split; aim for ≥85% Accuracy and ≥0.82 Macro-F1 on the seed dataset.

4.Explainable Insights (Top Terms & Samples)

Expose class-wise top features (e.g., LR coefficients) and show 3 example reviews per class so users can understand why a label was assigned.

5.Aspect/Theme Mining for Actionability

Auto-extract frequent themes from negative reviews (e.g., crash, battery, login, UX) via n-grams/keywords and cluster small phrases; ensure each theme shows frequency, sample quotes, and trend.

6.Actionable Recommendation Engine

Convert themes into developer tasks with a simple scoring: severity = frequency × negativity × recency. Display a Top-5 prioritized fix list with one-line "what to do" suggestions.

7.Interactive Analyst Dashboard

Build a local dashboard with filters (App, Date Range, Rating, Sentiment, Version) and live charts: Pie (distribution), Bar (by app/version), Line (weekly trend). Target <500ms UI refresh for typical queries.

III.PROPOSED SYSTEM

The proposed system, "AI-Powered Visualization & Recommendation of Multi-App Reviews," is designed as an offline, end-to-end framework that ingests user review datasets, processes them with NLP and ML, and provides actionable insights through an interactive visualization dashboard. The system eliminates the need for third-party APIs or cloud services, ensuring privacy, portability, and cost-effectiveness

- **Completely Offline** – no internet, no third-party APIs.
- **Lightweight & Portable** – SQLite + Python ensures easy deployment.
- **Interactive** – modern GUI for non-technical users.
- **Action-Oriented** – recommendations to improve app quality.
- **Extendable** – future scope for deep learning, multilingual support, cross-app comparisons.
- Relevant mathematical model associated with the Project
- **Input reviews** → Vector space (TF-IDF) → Classification model (Logistic Regression/SVM) → Sentiment labels.
- **Negative reviews** → Keyword extraction + Clustering → Themes.
- **Themes** → Severity scoring → Prioritized Recommendations.
- **Results** → Stored in SQLite → Rendered on NiceGUI dashboard.

IV.OUTCOMES

The proposed project "AI-Powered Visualization & Recommendation of Multi-App Reviews" is expected to deliver the following outcomes:

1. Offline Review Analytics System – A fully functional platform for analyzing app reviews without relying on third-party APIs or cloud services.

2.Automated Sentiment Classification –Reviews categorized into Positive, Neutral, and Negative sentiments using machine learning models.

3.Interactive Visualization Dashboard–Graphical insights through pie charts, bar charts, and time-trend graphs accessible via NiceGUI.

4.Theme & Keyword Extraction–Identification of frequently occurring issues (e.g., crash, battery drain, login problems, UI complaints).

5.Actionable Recommendations –Prioritized improvement suggestions for developers derived from user feedback.

6.Cross-Version & Multi-App Comparison – Ability to compare sentiment trends between different apps or app versions.

7.Exportable Reports – Generation of structured PDF/CSV reports containing analytics and recommendations.

8.Portable & Lightweight Framework – SQLite-based backend ensuring easy deployment on standard laptops/desktops.

V.CONCLUSION

The proposed system "AI-Powered Visualization & Recommendation of Multi-App Reviews" successfully addresses the challenge of converting unstructured user feedback into structured, actionable insights. By integrating NLP preprocessing, sentiment classification, keyword extraction, visualization, and recommendation generation within a single offline framework, the project enables developers to understand user perceptions and prioritize improvements effectively.

The use of Python, SQLite, and NiceGUI ensures that the solution remains lightweight, portable, and independent of third-party APIs, making it suitable for academic as well as industrial applications. The dashboard not only provides sentiment distribution and trend analysis but also delivers developer-oriented recommendations, bridging the gap between raw feedback and concrete action plans.

VI.FUTURE SCOPE

While the current system provides a robust foundation, it can be extended in the following ways:

1.Deep Learning Models: Integration of BiLSTM or Transformer models (e.g., BERT) to further improve sentiment classification accuracy.

2.Aspect-Level Analysis: Extracting fine-grained insights (UI, performance, security, battery) for more targeted recommendations.

3.Multilingual Support: Expanding NLP pipelines to process reviews in multiple languages beyond English.

4.Explainability (XAI): Adding keyword highlighting and interpretable AI methods to show why a review is classified in a particular category.

5.Scalability: Supporting large datasets (millions of reviews) through optimized storage and batch processing.

6.Integration with CI/CD: Linking insights directly with development pipelines for automated bug triaging and feature planning.

Cross-Domain Adaptation: Applying the same framework to analyze product reviews, e-learning feedback, or healthcare applications

VII.REFEERENCE

[1] N. Mittal and A. Chauhan, "A survey with sentiment analysis of online reviews," J. King Saud Univ. - Comput. Inf. Sci., vol. 30, no. 4, pp. 557–575, 2018.

[2] M. N. Hossain, X. Y. Wang, J. H. Lee, and K. H. Choi, "Mobile app review classification and analysis for mobile app development: A survey," IEEE Access, vol. 8, pp. 111547–111580, 2020.

[3] C. Hu, Y. Li, and M. Wang, "Review quality prediction via neural networks and user behavior features," in Proc. IEEE Int. Conf. Data Eng. (ICDE), 2019, pp. 1606–1617.

[4] S. Mukherjee, S. Bhowmick, and A. Ganguly, "Aspect-based opinion mining from user reviews around applications," Inf. Process. Manage., vol. 57, no. 1, 2020.

[5] Y. Zhang, K. Huang, Z. Li, and K. Wang, "Deep learning for aspect-based opinion mining on app reviews," in Proc. ACM WebSci Conf., 2021, pp. 49–58.

[6] L. Liu, F. Xie, Y. Chen, and X. Li, "Towards multi-level sentiment analysis for app reviews using transformer models," IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 6, pp. 2479–2491, 2021.

[7] J. Lin, Z. Ding, and X. Wu, "Cross-platform app review mining for mobile application analytics," in Proc. IEEE Int. Conf. Mobile Data Manage. (MDM), 2022, pp. 123–132.

[8] H. Liu, J. Yang, Y. Sun, and B. Li, "Recommendation of app improvements based on review mining," in Proc. IEEE Int. Conf. Softw. Qual., Rel. Secur. (QRS), 2022, pp. 289–298.

[9] X. Xu, L. Pan, and Y. Gao, "A hybrid approach for automatic bug report generation from app reviews," IEEE Access, vol. 10, pp. 45678–45690, 2022.

[10] K. Saunshi, M. H. Lee, and S. Kale, "Multi-task learning on app reviews for sentiment, topic, and rating prediction," in Proc. AAAI Conf. Artif. Intell., 2023.

[11] Y. Chen, S. Xu, and T. Zhao, "Explainable sentiment analysis for mobile app reviews: Benchmark and methods," Inf. Process. Manage., vol. 61, no. 3, 2024.

[12] A. Verma and R. K. Singh, "Graph-based recommendation of software improvements from user feedback," in Proc. IEEE Int. Conf. Big Data (BigData), 2024, pp. 789–798.

[13] P. Zhang, Z. Xu, and W. Wu, "Sentiment trend visualization for app feedback analytics," IEEE Trans. Vis. Comput. Graph., vol. 30, no. 1, pp. 345–358, 2024.

[14] B. Kim, S. Lee, and J. Park, "On privacy-preserving review analysis: Local differential privacy in user feedback," in Proc. IEEE Symp. Secur. Privacy (SP), 2025, pp. 112–124.

[15] M. Singh, V. K. Bansal, and R. Kumar, "Offline sentiment mining framework for app reviews without third-party APIs," IEEE Access, 2025, to appear.