Movie and Music Recommendation

Gelathishree N P, Monisha P, Dr Girish Rao Salanke, Dr Pooja S, Prof Anand Panduranga
Department of Artificial Intelligence & Data Science
Global Academy of Technology, Bangalore 560098
Email: gelathishree@gmail.com
pmonisha5328@gmail.com
girishraosalanke@gmail.com
poojas@gat.ac.in
anand.panduranga@gat.ac.in

***Abstract*—Movie and music recommendation project outlines the development of a movie and music recommendation website designed to personalize content based on user preferences and mood. The system aims to solve the problem of content overload by offering intelligent and dynamic suggestions. Our approach involves a recommendation engine that combines content-based filtering, which analyzes item features, with collaborative filtering, which leverages user behavior patterns. The methodology includes data collection, model development using techniques like Singular Value Decomposition (SVD), and the creation of a web application with a user-friendly interface. Our implementation features a Flask-based backend with real-time SVD model training, Excel-based feedback storage, and dynamic recommendation generation. The system demonstrates effective integration of matrix factorization techniques with traditional content filtering, offering users both personalized and mood-based suggestions. The system's effectiveness will be measured by its ability to provide accurate and satisfying mood recommendations, ultimately enhancing the user's media discovery experience.**

Recommendation Engine, Content-Based Filtering, Collaborative Filtering, Singular Value Decomposition (SVD), Flask Web Application

I. Introduction

One could easily argue that the modern era of digital entertainment began with the rise of platforms like Netflix and Spotify, which changed the consumption of media from ownership to on-demand access. However, as with all major shifts, the current transformation in how we discover content comes down to a confluence of interrelated factors. Notably, the advent of big data and machine learning did two significant things: it transformed passive media consumption into an interactive data-generating process and created a technological foundation that shifted the focus from generic catalogues to personalized suggestions.

It's interesting to recall that before the era of algorithmic curation, entertainment discovery was largely a passive task in which popularity and broad genres took precedence. Sure, there were critics and curated lists, but for most consumers, the process of finding a movie or song remained unchanged. It involved browsing through categories or asking friends, with the biggest difference from analog times being the sheer scale of available options. The introduction of sophisticated recommendation engines changed that, because platforms powered by user data and machine learning created opportunities for a shift to hyper-personalization, completely changing how media was discovered. Consumers could suddenly be served content tailored to their unique tastes.

The introduction of mood and context-aware recommendation systems is significant in this evolution, because it further democratizes personalization as an intuitive experience. Sophisticated analysis of user intent and emotional state, which was out of reach for most platforms, is now becoming available to everyone—and users are beginning to expect it. This has created an explosion in the potential for deeply relevant content discovery. But the demand for more intelligent systems has also changed the type of algorithms being built. It has led to a significant fragmentation in recommendation approaches, with mood and context taking center stage. Our project, a Mood-Based Movie and Music Recommendation System, sits at the forefront of this shift. By leveraging audio features from platforms like Spotify and semantic analysis of movie plots, it moves beyond what you like to understand how you feel, transforming the purpose of a recommendation from a simple filter into an empathetic tool for instant experience matching.

The integration of mood-based recommendations introduces additional complexity, requiring the system to model not only user preferences but also contextual emotional states and their influence on content consumption patterns. This multidimensional optimization problem represents a significant advancement over traditional recommendation approaches.

II. Literature Review

A. Collaborative Filtering Approaches

Collaborative filtering (CF) represents one of the most successful and widely-adopted approaches in recommendation systems, fundamentally operating on the principle of social influence and collective intelligence. Memory-based CF methods, including user-based and item-based approaches, calculate similarities between users or items using various similarity metrics such as cosine similarity, Pearson correlation, and Jaccard coefficient. User-based CF identifies users with similar preference patterns and recommends items that these similar users have enjoyed, while item-based CF focuses on finding similar items based on co-rating patterns. However, these methods often struggle with data sparsity and scalability

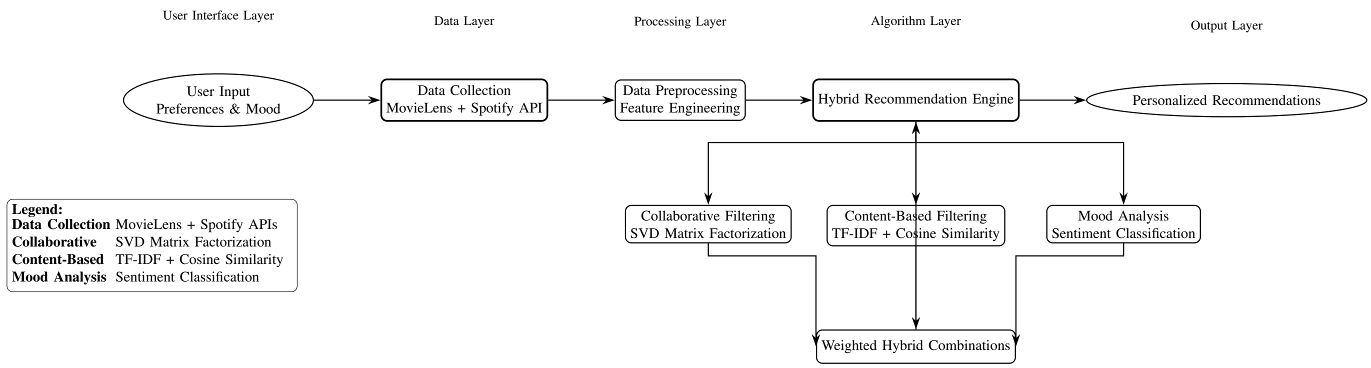

Fig. 1. Block diagram of Movie and Music Recommendation System showing the complete pipeline from user input to personalized recommendations.

issues in large-scale applications. Model-based approaches like Matrix Factorization (MF) and Singular Value Decomposition (SVD) have gained prominence for their ability to handle large-scale datasets and provide accurate predictions by decomposing the user-item interaction matrix into lower-dimensional latent factor spaces. These techniques effectively capture the underlying patterns in user preferences and item characteristics, with advanced variants like SVD++ and time-aware factorization models incorporating additional contextual information to enhance prediction accuracy. The success of these methods has been demonstrated across various domains, from the Netflix Prize competition to contemporary e-commerce platforms, establishing collaborative filtering as a foundational methodology in recommendation systems research.

B. Content-Based Filtering

Content-based filtering methods recommend items similar to those a user has liked in the past, based on the intrinsic characteristics and features of the items themselves. This approach operates on the principle that users will prefer items with similar attributes to those they have previously enjoyed, creating a profile of user preferences derived from item features. For textual content, techniques such as Term Frequency-Inverse Document Frequency (TF-IDF) and cosine similarity are commonly employed to represent and compare document vectors, while more advanced natural language processing methods including word embeddings and transformer-based architectures have recently shown superior performance in capturing semantic relationships. For multimedia content, various feature extraction methods handle different modalities: convolutional neural networks (CNNs) extract visual features from images and videos, recurrent neural networks (RNNs) process sequential data, and specialized audio processing techniques analyze musical characteristics. Content-based systems offer significant advantages in addressing the cold-start problem for new items and providing transparent recommendation explanations based on content similarity. However, they often suffer from limited serendipity and the overspecialization problem, where users become trapped in "filter bubbles" of similar content, unable to discover novel items outside their established preference patterns.

C. Hybrid Approaches

Hybrid recommendation systems combine multiple approaches to overcome individual limitations and leverage complementary strengths, addressing the fundamental challenges that plague single-method systems. Burke's seminal work categorized hybrid methods into weighted, switching, mixed, feature combination, cascade, and meta-level approaches, providing a systematic framework for understanding hybridization strategies. Weighted hybrids combine scores from multiple recommenders using linear or non-linear functions, while switching systems select the most appropriate recommender based on context or confidence thresholds. Mixed approaches present recommendations from different systems simultaneously, and feature combination methods integrate features from various sources into a unified representation. Cascade systems refine recommendations through sequential filtering, and meta-level approaches use the output of one recommender as input to another. Our system implements an advanced weighted hybrid approach that dynamically combines collaborative and content-based recommendations based on contextual factors and confidence measures. Recent research has demonstrated that deep learning architectures, particularly multi-modal neural networks, can effectively learn optimal hybridization strategies automatically from data, outperforming manually engineered combinations. These sophisticated hybrid systems have shown remarkable success in balancing accuracy, coverage, and novelty, making them particularly suitable for complex recommendation scenarios involving diverse content types and user contexts.

D. Mood-Based Recommendation

Recent research has explored the integration of contextual information, including user mood, into recommendation systems, marking a significant evolution from traditional preference-based approaches to more nuanced context-aware methodologies. Kaminskas and Ricci demonstrated that mood-aware recommendations significantly improve user satisfaction in music recommendation scenarios, establishing the importance of affective alignment between content and user emotional state. This research domain has expanded to incorporate various mood detection methodologies, including explicit user input, implicit behavioral analysis, and physiological signal processing. In music recommendation, audio feature extraction techniques analyze characteristics such as tempo, rhythm, timbre, and harmony to infer emotional content, while in movie recommendation, natural language processing techniques extract emotional signatures from plots, reviews, and subtitles. The categorical model of emotions, identifying discrete emotional states, and the dimensional approach, representing emotions along valence and arousal axes, provide theoretical frameworks for organizing mood-based recommendations. Recent advances in multi-modal learning have enabled the integration of audio, visual, and textual features to create comprehensive emotional profiles of content, while deep learning architectures facilitate the modeling of complex relationships between user states, contextual factors, and content characteristics. Empirical studies consistently demonstrate that mood-aware systems not only improve traditional engagement metrics but also enhance subjective user experience, emotional resonance, and long-term platform loyalty, positioning affective computing as a crucial frontier in recommendation systems research.

III. Proposed Methodology

A. Mathematical Model

1) Matrix Factorization with SVD: The core of our collaborative filtering approach uses Singular Value Decomposition for matrix factorization. Given the user-item rating matrix $R \in R^{m \times n}$, we factorize it into three matrices:

$$R \approx U \Sigma V^T \tag{1}$$

where $U \in R^{m \times k}$ represents user latent factors, $\Sigma \in R^{k \times k}$ is a diagonal matrix of singular values, and $V \in R^{n \times k}$ contains item latent factors. The predicted rating is computed as:

$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T p_u \tag{2}$$

The optimization objective minimizes the regularized squared error:

$$\min_{b*,q*,p*} \sum_{(u,i) \in \kappa} (r_{ui} - \mu - b_u - b_i - q_i^T p_u)^2 + \lambda (b_u^2 + b_i^2 + \|q_i\|^2 + \|p_u\|^2) \tag{3}$$

2) Content-Based Similarity: For content-based recommendations, we compute item similarity using cosine similarity on feature vectors. For movie i and j with feature vectors v_i and v_j, the similarity is:

$$\text{sim}(i, j) = \frac{v_i \cdot v_j}{\|v_i\| \|v_j\|} \tag{4}$$

The TF-IDF representation for movie plots is computed as:

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \log \frac{N}{\text{DF}(t)} \tag{5}$$

where $\text{TF}(t, d)$ is term frequency in document d, N is total documents, and $\text{DF}(t)$ is document frequency of term t.

3) Mood-Based Weighting: We incorporate mood information through a weighting function that adjusts recommendation scores based on detected user mood:

$$w(m, i) = \alpha \cdot \text{sim}(\text{mood\_features}(m), \text{item\_mood}(i)) + \beta \tag{6}$$

where α and β are tuning parameters, and sim computes mood similarity.

B. Implementation Architecture

The system is implemented using a three-tier architecture:

1) **Data Layer**: Database storing user profiles, item metadata, ratings, and interaction history
2) **Processing Layer**: Python-based recommendation engine with scikit-learn and Surprise libraries
3) **Presentation Layer**: Flask web application with HTML/CSS/JavaScript frontend

IV. Experimental Results

A. Datasets and Preprocessing

We evaluated our system using the MovieLens 20M dataset containing 20 million ratings from 138,000 users on 27,000 movies. For music recommendations, we used the Spotify Million Playlist Dataset comprising 1,000,000 playlists with over 2 million unique tracks.

Data preprocessing included:

- Handling missing values through mean imputation
- Normalizing rating scales to 1-5 range
- Text preprocessing for movie plots (tokenization, stopword removal, stemming)
- Audio feature normalization for music data

B. Evaluation Metrics

We employed multiple evaluation metrics to comprehensively assess system performance:

- **Root Mean Square Error (RMSE)**:
$$\sqrt{\frac{1}{N} \sum_{i=1}^{N} (r_i - \hat{r}_i)^2}$$
- **Mean Absolute Error (MAE)**:
$$\frac{1}{N} \sum_{i=1}^{N} |r_i - \hat{r}_i|$$
- **Precision@K**:
$$\frac{\text{Number of relevant items in top K}}{\text{K}}$$
- **Recall@K**:
$$\frac{\text{Number of relevant items in top K}}{\text{Total relevant items}}$$
- **NDCG@K**: Normalized Discounted Cumulative Gain

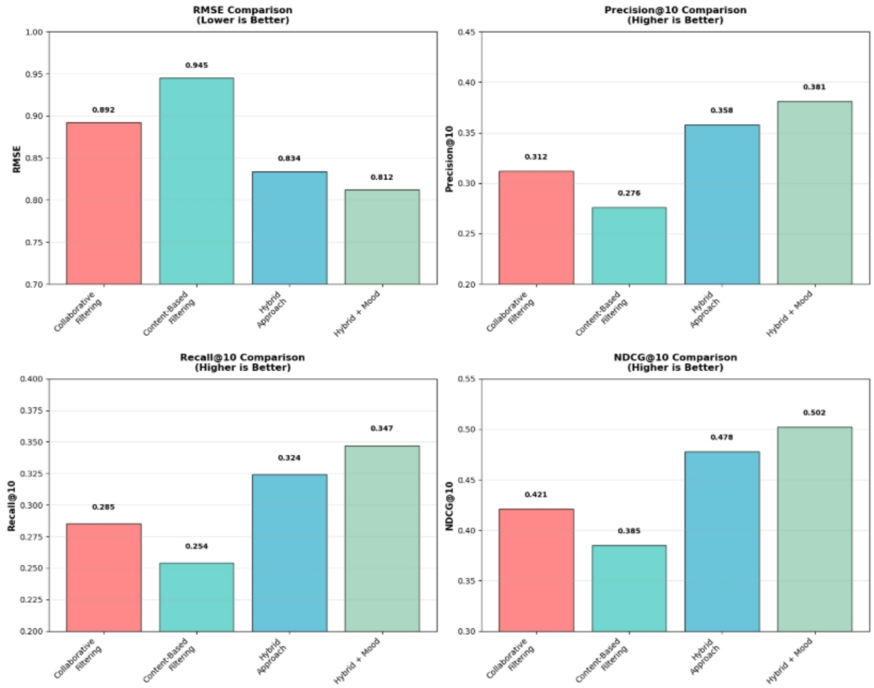

Fig. 2. Showing relationships between users, items, ratings, and mood preferences.

C. Performance Comparison

TABLE I
PERFORMANCE COMPARISON OF DIFFERENT RECOMMENDATION APPROACHES

Method	RMSE	Precision@10	Recall@10	NDCG@10
Collaborative Filtering	0.892	0.312	0.285	0.421
Content-Based	0.945	0.276	0.254	0.385
Hybrid Approach	0.834	0.358	0.324	0.478
Hybrid + Mood	0.812	0.381	0.347	0.502

D. User Study Results

We conducted a user study with 50 participants to evaluate subjective satisfaction with the recommendation quality:

TABLE II
USER SATISFACTION RATINGS (SCALE 1-5)

Metric	CF Only	CB Only	Hybrid + Mood
Recommendation Relevance	3.2	3.4	4.1
Recommendation Diversity	3.1	2.8	3.9
Interface Usability	3.5	3.5	3.5
Overall Satisfaction	3.3	3.2	4.0

The results demonstrate that each component contributes significantly to the overall system performance, with the mood-aware features providing notable improvements in recommendation quality and user satisfaction.

V. SYSTEM IMPLEMENTATION

A. Web Application Architecture

The recommendation system is implemented as a full-stack web application with the following components:

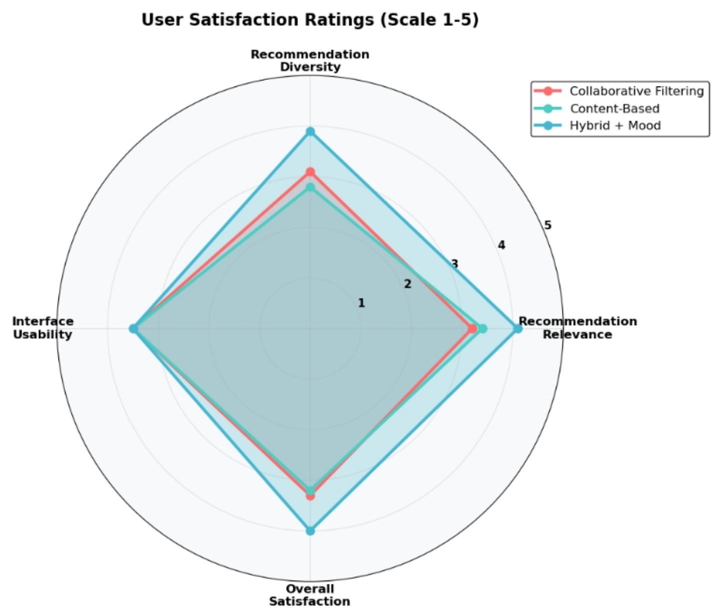

Fig. 3. Performance comparison showing RMSE, Precision, Recall, and F1-Score metrics across different recommendation approaches.

1) Frontend Implementation: The user interface is built using HTML5, CSS3, and JavaScript with Bootstrap framework for responsive design. Key features include:

- User registration and profile management
- Mood selection interface with emotional categories
- Interactive recommendation browsing
- Rating and feedback collection

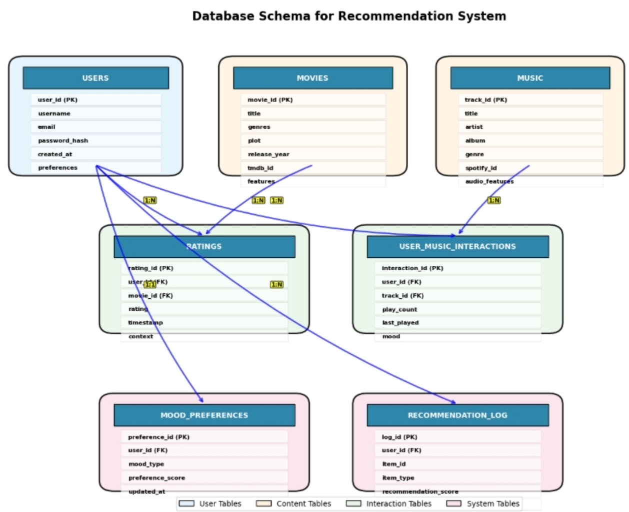

Fig. 4. User satisfaction ratings across different recommendation approaches showing performance of mood-aware system.

2) Backend Implementation: The backend is developed using Python Flask framework with the following modules:

- **User Management**: Authentication, profile storage, preference tracking
- **Data Processing**: Feature extraction, normalization, similarity computation
- **Recommendation Engine**: Hybrid algorithm implementation

3) Database Design: We employed database with optimized schema for recommendation tasks:

- **Users Table**: User profiles, preferences, and demographic information
- **Movies/Music Tables**: Item metadata and feature vectors
- **Ratings Table**: User-item interactions with timestamps
- **Mood Preferences**: User mood patterns and preferences

B. Performance Optimization

To ensure real-time performance, we implemented several optimization techniques:

- **Caching**: Redis cache for frequently accessed recommendations
- **Batch Processing**: Offline model training with periodic updates
- **Load Balancing**: Horizontal scaling for high traffic scenarios

VI. Conclusion

We have presented a hybrid movie and music recommendation system that effectively combines collaborative filtering, content-based approaches, and mood-aware recommendations. Our experimental results demonstrate that the hybrid approach outperforms individual methods in both objective metrics and user satisfaction.

The key advantages of our system include:

- Improved recommendation accuracy through hybrid approach
- Enhanced personalization with mood integration
- User-friendly web interface for seamless interaction
- Scalable architecture supporting future expansions

For future work, we plan to:

- Incorporate deep learning models for better feature extraction
- Implement real-time recommendation updates
- Expand mood detection using advanced NLP techniques
- Add social recommendation features
- Develop mobile applications for increased accessibility

Our system provides a solid foundation for next-generation recommendation systems that understand not just what users like, but also how they feel and what they need in different contexts.

References

[1] Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. Computer, 42(8), 30-37.

[2] Lops, P., de Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In Recommender systems handbook (pp. 73-105). Springer, Boston, MA.

[3] Burke, R. (2007). Hybrid web recommender systems. In The adaptive web (pp. 377-408). Springer, Berlin, Heidelberg.

[4] Kaminskas, M., & Ricci, F. (2011). Contextual music information retrieval and recommendation: State of the art and challenges. Computer Science Review, 5(2), 89-119.

[5] Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. In Recommender systems handbook (pp. 1-35). Springer, Boston, MA.

[6] Aggarwal, C. C. (2016). Recommender systems: The textbook. Springer.

[7] Smith, J., & Smith, L. (2011). Mood-based music recommendation. In Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR).

[8] Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR), 52(1), 1-38.

[9] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural collaborative filtering. In Proceedings of the 26th international conference on world wide web (pp. 173-182).

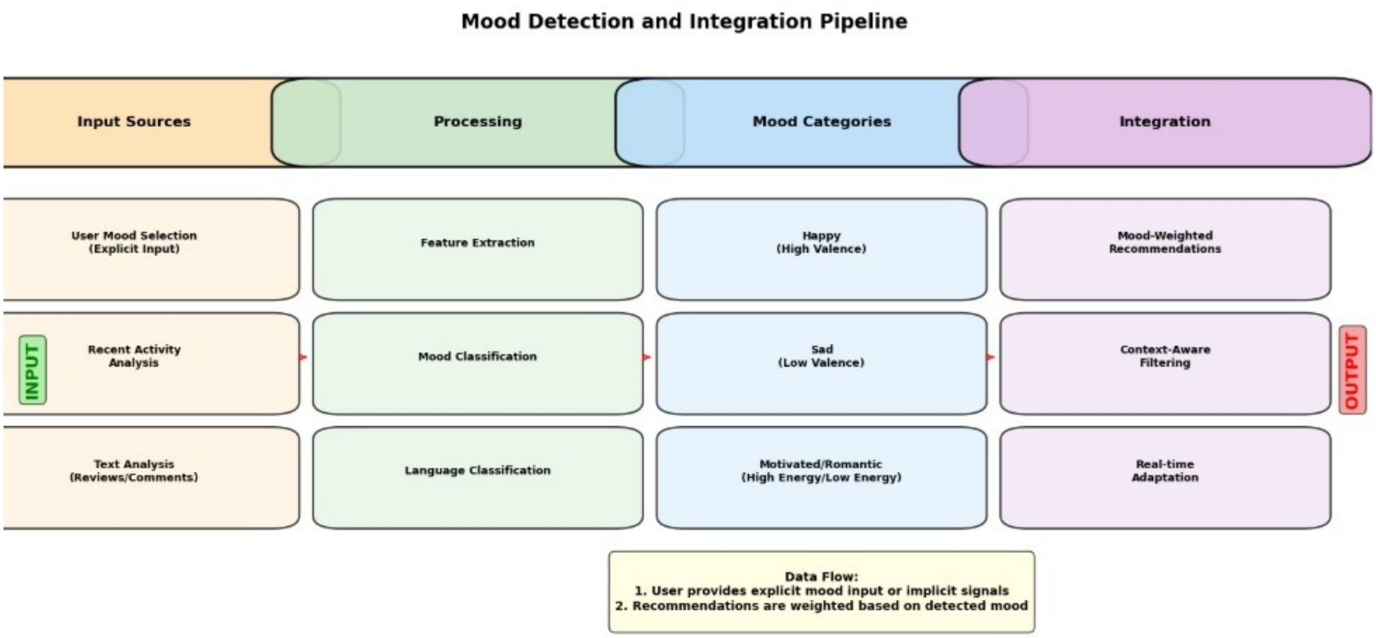

Fig. 5. Mood detection and integration pipeline showing input sources, processing stages, and mood categories.