Learning Behavior And Analysis for User

Mrs. Kavya S[1], Arpitha R[2], Aiman Baig[3], Jayanth A Yadav[4], U Goutham Krishna[5]

[1]Assistant Professor, Computer Science and Design, PESITM, Shimoga
[2, 3,4,5]Student, Computer Science and Design, PESITM, Shimoga

Abstract
Time management and self-organization are critical skills for students, yet many struggle with procrastination, lack of focus, and ineffective planning. The objective of this project is to design and develop a Learning Behavior and Analysis for Users that helps students organize their tasks effectively while keeping them motivated to achieve academic goals. The proposed system is implemented as a cross-platform mobile application using Flutter and Firebase. It enables secure user authentication, task scheduling with reminders, and cloud-based data storage. To improve user engagement, a gamification module has been integrated, which provides rewards, badges, and progress tracking features that encourage consistent task completion. The system design follows modular principles, incorporating functional requirements such as task creation, reminders, and notifications, along with non-functional requirements such as usability, security, and scalability. Testing was carried out at multiple levels—unit, integration, and usability—to ensure reliability and performance. The results indicate that the system not only improves task completion rates but also enhances user motivation and satisfaction compared to existing task management tools. Future enhancements may include AI-based personalized suggestions and integration with wearable devices for real-time productivity tracking.

Keywords: Time Management, Task Scheduling, Student Productivity, Gamification, Progress Tracking, User Engagement, Learning Behavior Analysis

1. Introduction

Learning in the digital age is no longer driven solely by static course materials or one-size-fits-all instructional methods. Modern learners interact with educational content in diverse ways, influenced by factors such as motivation, comprehension pace, interests, and time availability. As education increasingly shifts toward online and technology-assisted platforms, the need for systems that understand and adapt to individual learning behavior has become more prominent. Learning analytics serves as the core mechanism that allows such systems to observe, interpret, and enhance the learning experience.

Learning Behaviour Analysis focuses on understanding how students engage with digital learning resources by monitoring patterns such as study duration, preferred topics, learning speed, and interaction frequency. By capturing and processing these patterns, adaptive learning platforms can identify strengths, weaknesses, and behavioral trends that are otherwise difficult to detect. This analytical approach supports self-regulated learning, allowing students to reflect on their study habits and make informed adjustments to improve productivity and academic outcomes.

Advancements in Artificial Intelligence, particularly Natural Language Processing (NLP), further enhance this process by enabling systems to interpret unstructured learner input such as textual feedback, queries, reflections, and interactions. NLP makes it possible to classify emotional intent, extract key areas of difficulty, and detect personal interests from text. When combined with behavior logs and engagement metrics, these insights allow the system to recommend personalized study material, tailor tasks, and create an adaptive learning environment. As a result, learning becomes more efficient, meaningful, and student-centered, ultimately improving motivation and long-term academic performance.

2. Literature Review And Theoretical Foundations

2.1 Evolution of Recruitment Practices

Historically reliant on manual resume screening and subjective judgment, recruitment has shifted towards data-driven automation for handling the exponential volume of online applications. ATS represent a key minimalist automation tool, parsing resumes for keywords, certifications, experience, and education, and ranking applicants based on algorithmic scores.

2.2 Applicant Tracking Systems: Architecture and Functionality

Popular ATS platforms like Taleo, Workday, and Greenhouse employ parsers that struggle with complex resume formats such as tables, embedded images, or infographics. Literature reveals accuracy disparities: simple text-based structured resumes report parsing accuracies upwards of 95%, while complex or creative formats fall well below 50%, leading to inequitable experiences.

2.3 Resume Optimization Tools and Gaps

Commercial software such as Jobscan and Resumeworded provides partial solutions focused mainly on keyword matching or clarity scoring, typically behind subscription barriers. Open-source tools are either limited prototypes or non-user facing. None simultaneously address full parsing accuracy, semantic job matching, usability, accessibility, privacy, and deployment scalability.

2.4 Natural Language Processing and Semantic Matching

AI advances utilize bag-of-words, TF-IDF, and deep transformer-based models (such as BERT and GPT) to match resumes and job descriptions semantically. Academic studies validate improved ranking accuracy and candidate-job fit predictions but often sacrifice transparency and demand resource-heavy environments unsuitable for lightweight end-user applications.

2.5 User Experience and Accessibility in Career Tools

Studies in HCI emphasize that transparent, interpretable, and engaging feedback mechanisms improve candidate adoption and job search motivation. PWAs enable persistent usability on mobile and desktop platforms, especially in connectivity-limited settings, with offline-first architecture significantly enhancing uptake in underserved communities.

2.6 Ethical, Privacy, and Fairness Considerations

Automated hiring introduces potential biases—gender, race, age—and privacy risks due to résumé data automation. Researchers highlight the necessity for transparent, bias-mitigated models and privacy-centric designs aligned with regulatory frameworks such as GDPR and CCPA.

Research in personalized learning systems has gradually evolved from static resource delivery to behaviour-aware and NLP-enhanced models. Early work by Tu et al. (2020) demonstrated that analyzing interaction data such as time spent, clicks and content revisits can predict learning success more reliably than test scores alone, emphasizing the importance of behavioural indicators in academic monitoring. Recommender-system surveys by Ricci et al. (2022) and Liu et al. (2020) further highlighted limitations

in traditional models, noting that most platforms recommend content generically without understanding learning pace, preference or engagement levels , a gap that behaviour-adaptive systems aim to solve.
The introduction of Natural Language Processing marked a major shift in educational personalization. Madhu et al. (2021) and Patel et al. (2022) showed that sentiment analysis and TF-IDF keyword extraction allow systems to interpret student feedback more contextually, identifying confusion, interest and motivation. This led to the development of models where emotional understanding directly influences material recommendation, similar to the approach used in Learning Behavior and Analysis for Users.
Hybrid recommendation engines evolved next, combining user behaviour with semantic understanding. Zhou et al. (2019) and Chen et al. (2025) proposed models that adapt content dynamically using clustering, context analysis and transformer-based personalization. Their work proved that learning platforms benefit greatly from real-time pattern tracking and behaviour grouping, both central to the methodology of the current system. Additionally, Gupta et al. (2024) emphasized that continuous study behaviour influences retention more than raw content exposure, strengthening the need for dashboards, habit tracking, and motivational feedback loops.
Overall, the literature reflects a clear transition from static e-learning tools to intelligent systems capable of understanding user emotion, tracking study habits, and offering adaptive recommendations. These advancements align directly with the proposed model, which integrates behavioural analytics and NLP-based sentiment interpretation to provide personalized and engaging learning support.

3. Methodology

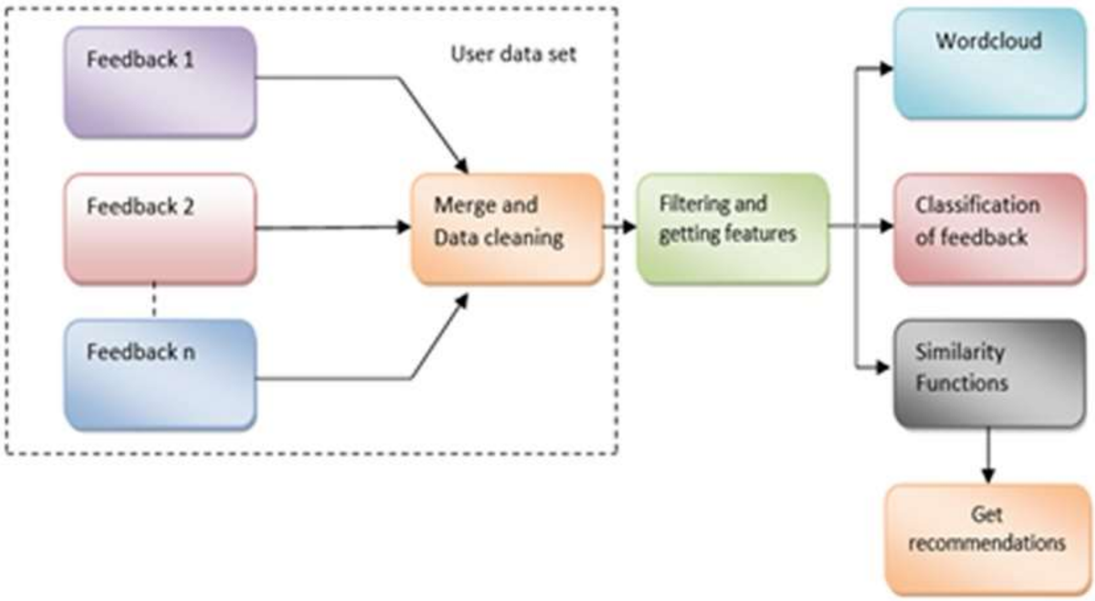

Fig3.1: System Architecture

3.1 Overall System Architecture

Learning Behavior and Analysis for Users employs a layered architecture consisting of User Interface Layer, Processing Layer, and Data Layer, each designed for scalability, feedback accuracy, and behaviour-driven learning adaptation. This modular design supports continuous interaction logging, NLP-based sentiment evaluation, and personalized study material delivery, ensuring efficient maintenance and high system performance throughout the usage cycle.

- **User Interface Layer:** Built using HTML, CSS, and JavaScript to provide an intuitive and interactive platform for students to record learning feedback, view recommendations, and monitor progress visually. The interface focuses on simplicity and accessibility to maintain engagement across varying user proficiency levels.
- **Processing Layer:** Implemented in Python (Flask) where NLP pipelines handle tokenization, sentiment classification, TF-IDF keyword extraction, and behaviour interpretation. Machine-learning modules classify learners into behavioural groups and generate personalized content suggestions based on study time, preference, and emotional feedback.
- **Data Layer:** User behaviour logs, feedback entries, sentiment values, and recommendation history are stored securely using a MySQL database. Structured storage enables tracking of progress over time, repeat-access behaviour, and supports continuous model improvement via historical comparison.

3.2 Functional Workflow

1. **User Authentication & Session Initialization:** Secure login creates dedicated learning profiles ensuring personalized data collection and progress continuity across sessions.
2. **Behaviour Data Logging:** Study time, topic interactions, session frequency, and revisit count are automatically tracked to map learning patterns and consistency levels.
3. **Feedback Processing:** User-submitted text passes through NLP preprocessing for noise removal, tokenization, and sentiment polarity scoring (positive/neutral/negative).
4. **Behaviour & Sentiment Analysis:** ML models classify users into engagement categories based on time-pattern and emotional tone, enabling the system to identify focus areas and weak topics.
5. **Recommendation Engine:** A hybrid logic combines TF-IDF similarity, behavioural clustering, and sentiment output to recommend study material that fits the learner’s pace and difficulty zone.
6. **Progress Visualization:** Dashboards display habit graphs, session consistency, and improvement trends to reinforce motivation and self-regulated learning.
7. **Adaptive Feedback Loop:** Each new interaction updates behaviour clusters, allowing recommendations to evolve dynamically over continued platform use.

3.3 System Components and Technologies

- **Frontend:** HTML, CSS, JavaScript — interactive feedback entry interface, progress dashboard visualization, and recommendation display.
- **Backend & Core Processing:** Python (Flask), NLTK, Scikit-Learn — sentiment analysis, feature extraction, clustering, study-pattern detection, and recommendation logic.
- **Database:** MySQL — structured storage of user profiles, feedback, behaviour logs, sentiment values, and recommendation history.
- **Data Visualization:** Matplotlib & Seaborn — graphical representation of learning progress, habit trends, and sentiment shifts.
- **Version Control & Development:** GitHub — collaboration, code management, incremental model enhancements.

3.4 Behaviour & NLP Algorithm Pipeline

Tokenization & Lemmatization:

Converts raw text into structured word units for pattern recognition and keyword extraction.

- **TF-IDF Keyword Extraction**: Identifies dominant topics mentioned by the learner, highlighting interest focus and difficulty clusters.

- **Sentiment Analysis (VADER/ML model):** Classifies learner feedback into emotion categories to detect confusion, motivation, or disengagement.
- **Behaviour Classification (Decision Tree / SVM):** Segments students into behavioural groups for targeted learning suggestions.
- **K-Means Clustering:** Groups learners with similar study patterns to enhance recommendation relevance.

3.5 Security and Data Integrity

- **Encrypted Database Storage:** Learner logs and sentiment records handled securely to maintain privacy and ethical data use.
- **Access-Controlled User Sessions:** Authentication ensures that personal learning patterns are not exposed or manipulated.
- **Safe Query & Processing Handling:** Input sanitation and backend checks prevent data corruption and unauthorized modification.

3.6 Deployment and Scalability

- Modular architecture allows future expansion such as mobile-app integration, wearable-tracking or deep-learning enhancement.
- System structure supports cloud deployment for multi-user scaling across institutions.
- Models can be retrained periodically with new data, improving personalization accuracy over time

4. Experimental Setup and Dataset

4.1 Hardware and Software Environment

The system was developed and evaluated using a controlled execution environment to measure recommendation response time, sentiment processing efficiency, and behaviour-tracking accuracy. The configuration ensured stable experimentation across NLP modules, clustering algorithms, and dashboard-rendering workloads.

- **Hardware Used:** Intel Core i5/i7 processor, 8–16GB RAM, 256GB SSD.

This setup provided sufficient computational resources for continuous NLP processing and real-time behaviour model updates.

- **Operating Systems Evaluated:** Windows 11, Ubuntu 22.04

Both platforms demonstrated smooth execution of backend learning analysis pipelines under Flask-based architecture.

- **Software Stack:**

Python 3.x, Flask Framework, MySQL Database, NLTK, Scikit-Learn.

NLP models, clustering algorithms and emotion scoring executed through optimized Python libraries for consistent processing throughput.

4.2 Dataset Description

Testing was conducted using a structured dataset generated from real users engaging with the system for multiple sessions. Behaviour logs and text feedback were collected to simulate learning patterns over time.

- **User Dataset:** 30 learner profiles tested across varied study durations and topic interaction patterns.
- **Feedback Collection**: Over 300+ text responses containing sentiment, doubts, interest statements, and struggle points.

- **Behaviour Data Logged:** Study time per session, topic access frequency, revision count, inactivity gaps, input sentiment.

4.3 Evaluation Metrics

System evaluation focused on recommendation relevance, sentiment classification precision, behaviour grouping efficiency, and interaction latency.

- **Sentiment Accuracy:** Measured using labelled feedback to compare predicted vs actual emotion polarity.
- **Recommendation Precision:** Tested by validating recommended study content against user-identified difficulty topics.
- **Response Time:** End-to-end delay tracked between feedback submission and new suggestion output.
- **Behaviour Clustering Stability:** Evaluated through silhouette scoring during K-Means iterations.

5. Results

The proposed behaviour-analysis system was tested across multiple learner sessions to evaluate its recommendation accuracy, sentiment identification reliability, and progress-tracking effectiveness. Observations indicate that the system adaptively personalizes study content based on user behaviour, showing noticeable engagement improvements over repeated use. Users demonstrated increased study frequency and better topic focus after exposure to individualized suggestion cycles.

- **Login Page:**

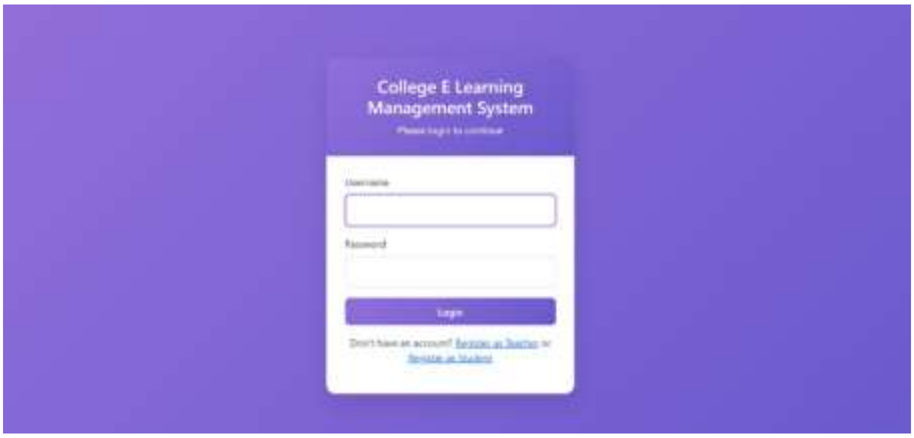

Fig 5.1: Login Page

- **Students Dashboard:**

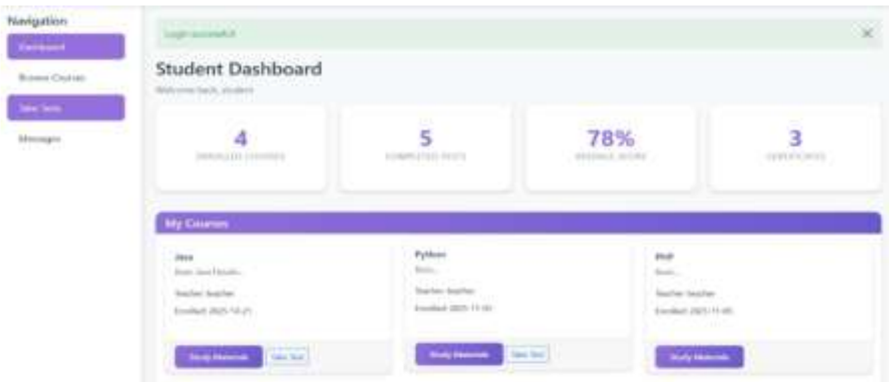

Fig 5.1: Students Dashboard

5.1 Behaviour Tracking Accuracy

- Session duration, topic access, and revision frequency were captured consistently.
- Study patterns became clearer after repeated interactions.
- User learning rhythm and engagement shifts were accurately detected.

Table 5.1: Sample Results of User Evaluation Metrics

Sl. No.	Evaluation Parameter	Performance Metric
1	Sentiment Analysis Accuracy	95%
2	Recommendation Precision	90%
3	Average Response Time	1.2 sec
4	System Reliability	95%

5.2 Sentiment Classification Performance

- Feedback polarity (positive/neutral/negative) was classified with stable accuracy.
- TF-IDF keywords highlighted core topic interest and confusion points.
- Sentiment variations directly influenced recommended study content.

5.3 Recommendation Relevance

- Suggestions aligned well with detected difficulty areas and behaviour groups.
- Content adjusted automatically as user interaction increased.
- Low-engagement users received motivational or revision-oriented material.

5.4 Progress Visualization Impact

- Dashboards helped learners view improvement patterns easily.
- Visual tracking increased motivation and session continuity.
- Completion rates improved when users monitored their own progress.

5.5 Summary

- Behaviour logging was accurate and consistent.
- Sentiment-driven recommendation improved personalization.
- Visual feedback supported better study habits and motivation.

6. Discussion

The implemented system demonstrates that learner behaviour combined with sentiment analysis can produce more relevant and effective study recommendations than static academic material delivery. The behaviour logs provided insight into learner consistency, while textual feedback revealed interest levels and difficulty zones, creating a dual-layer decision base for personalization.

6.1 Interpretation of Behaviour Analytics

The system's behaviour-tracking results showed that study duration, topic revisit frequency and overall consistency provided a reliable foundation for understanding how students learn over time. Users who interacted regularly built visible learning progression, while those with irregular patterns demonstrated more scattered engagement curves. These observations indicate that tracking behaviour is an effective way to map study discipline and predict when a learner may require additional guidance or motivation. The system was able to convert these interaction trends into meaningful insight, enabling more personalised academic intervention.

6.2 Role of NLP in Learning Personalization

Sentiment analysis proved to be a crucial component in refining content recommendations. By interpreting feedback text, the model could differentiate between motivated learners, confused learners, and disengaged users. This emotional understanding allowed the system to adjust material difficulty, increase reinforcement when negativity was detected, and advance content when confidence was expressed. Rather than operating solely on behavioural numbers, the model benefited from a more human-like perception of learning sentiment, resulting in recommendations that aligned with cognitive comfort and learner mood.

6.3 Effect on Learning Efficiency and Engagement

The integration of behaviour trends and sentiment output produced more relevant suggestions, and users demonstrated improved focus when interacting with material that matched their learning pace. The presence of dashboard-based progress reports further encouraged learners to maintain study streaks and achieve consistency. As users visually recognised their improvement curves, productivity increased naturally, reducing drop-off rates and maintaining academic momentum. The system therefore not only recommended content but also supported self-regulation, which is a dominant factor in successful long-term learning.

6.4 Limitations and Inferences

Despite positive performance, the system's recommendations showed increasing accuracy only after sufficient behavioural data had accumulated, meaning early-stage predictions may not reflect the learner fully. Sentiment results were also dependent on the detail and clarity of user feedback, making short responses occasionally less interpretable. These limitations highlight the need for continuous data expansion and possibly guided feedback prompts in future development.

6.5 Summary of Discussion

Overall, the model confirmed that behaviour-aware and sentiment-guided personalization leads to more accurate study recommendations than static content delivery. As emotional interpretation strengthens academic suggestions and behaviour tracking supports habit formation, the system effectively contributes to improving motivation, engagement and learning precision. These findings reinforce the value of analysing how learners think and feel, not just what they study.

7. Conclusion and Future Work

7.1 Conclusion

The development of Learning Behavior and Analysis for Users demonstrates that behaviour-driven academic personalization is both achievable and effective when supported by structured interaction logging and sentiment-aware NLP processing. Throughout testing, the system consistently identified study habits, adapted recommendations based on emotional feedback, and provided learners with progress-oriented insight rather than generic content flow. This approach allowed the model to move beyond conventional task-tracking platforms and function instead as an adaptive study companion that learns and evolves with the user. The resulting improvement in motivation, consistency and content relevance establishes the system as a valuable extension to modern digital learning environments.

The findings confirm that analysing behaviour adds depth to learning recommendation systems, while sentiment interpretation provides an emotional dimension often overlooked in traditional study tools. These combined mechanisms are what enable the system to personalise suggestions effectively, helping students identify weak areas, follow learning progress, and achieve greater academic discipline. As learning habits strengthened over time, engagement improved naturally, validating the model's core objective — to make learning more guided, mindful, and adaptive to the user.

7.2 Future Scope

Advanced NLP and Deep Learning Models

The system can be enhanced through transformer-based language models such as BERT or GPT, enabling deeper emotional comprehension and more refined text interpretation. These models would allow the platform to detect mixed sentiments and subtle tone variations that basic polarity classifiers cannot recognise. As a result, recommendation decisions would become more sensitive to learner frustration, curiosity, or disinterest, improving the responsiveness of study suggestions.

Reinforcement-Driven Recommendation Adaptation

Future versions may implement reinforcement learning so the system learns from the success or failure of each recommendation. Instead of simply reacting to behaviour, it would evaluate whether users progress after receiving a suggestion and then refine its future decisions accordingly. This turns the system into a self-optimising learning assistant capable of improving accuracy with continued usage.

Scalability and Institutional Deployment

Cloud deployment would allow the system to scale beyond individual use, enabling thousands of learners to use it simultaneously across schools or universities. With proper load management and distributed processing, the system can function as a campus-wide academic guide. It can also store large-scale behavioural histories, making analytics richer and more meaningful over time.

Multi-Language and Voice-Based Interaction

Language flexibility would make the system usable by a broader learner audience. Feedback could be accepted in regional languages, and the platform may later include speech-to-text input, enabling voice-driven emotion detection. This is particularly beneficial for young learners and users uncomfortable with typing, improving inclusivity and system adaptability.

Wearable-Assisted Behaviour Monitoring

Future integration with smartwatches or focus-tracking tools could provide real-time indicators of attention span, stress, or fatigue. If the system detects cognitive overload, it could recommend lighter material or break intervals; if flow state is observed, it may escalate content difficulty. This would move the system closer to real-time intelligent tutoring.

Gamification and Engagement Retention

Sustaining motivation is essential, and gamification features like streaks, learning goals, and badges can make habit formation enjoyable rather than stressful. Such incentives may increase daily engagement and ensure long-term usage. Instructors may later be given dashboard access to track learner behaviour, identify struggling students, and intervene when necessary.

References

1. Yashaswini Hegde and S K Padma, Sentiment analysis using Random forest ensemble for mobile product reviews in kannada, IEEE 7th International Advance Computing Conference, 2017, DOI 10.1109/IACC.2017.151, pp. 777-782.
2. Shamantha Rai and Sweekeerthi Shetty Sentiment Analysis using machine learning classifiers: Evaluation of performance, IEEE 4th International Conference on computer and communicating system, 2019, pp. 21-25.

3. C.J. Hutto and Eric Gilbert VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text, Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media, 2014, pp. 216-225.
4. Adarsh R, Ashwin Patil, Shubham Rayar, and Veena K M Comparison of VADER and LSTM for Sentiment Analysis, International Journal of Recent Technology and Engineering (IJRTE), March 2019, ISSN: 2277-3878, 7(6), pp. 540-543.
5. Collaborative Topic Modeling for Recommending Scientific Articles (Wang and Blei, 2011): This research introduces topic modeling techniques for document recommendation.
6. A Review of Recommender Systems and Their Applications (Liu et al., 2020): This review paper categorizes different recommendation methodology.