**BOOK RECOMMENDATIN USINNG AI**

**PRESENTED BY**

**STUDENT NAME:** SAURAV KUMAR GIRI

**COLLEGE NAME:** PURNEA COLLEGE OF

ENGINEERING, PURNEA

**DEPARTMENT**: COMPUTER SCIENCE

AND TECHNOLOGY (AI)

**EMAIL ID:** [sauravgiriofficial17@gmail.com](mailto:sauravgiriofficial17@gmail.com)

**AICTE REGISTRATION ID:**

STU680d325ee12dd1745695326

**OUTLINE**

1. Problem Statement
2. Proposed System/Solution
3. System Development Approach
4. Algorithm & Deployment
5. Result (Output Image)
6. Conclusion
7. Future Scope
8. References

**PROBLEM STATEMENT**

In today’s digital world, users face difficulty in selecting the right books from large libraries due to information overload. A system that understands user interests and provides relevant recommendations is needed to simplify the book discovery process and make it more personalized.

**PROPOSED SOLUTION**

The proposed system addresses the challenge of helping users discover relevant books by using AI and machine learning techniques. The solution is structured around the following key components:

**1. Data Collection**

* **Book metadata:** title, author, genre, summary
* **User data:** reading history, ratings, preferences
* **External sources:** Kaggle, Goodreads, and open book APIs

User history helps the system identify specific patterns in genre or author affinity, which is essential to delivering relevant recommendations.

**2. Data Preprocessing**

* Handle missing values in metadata fields
* Combine fields: title + author + genre + description
* Normalize text: lowercase, remove stopwords, punctuation, etc.
* Use **TF-IDF** to vectorize text content:

**TF-IDF(t, d) = TF(t, d) × log(N / DF(t))**

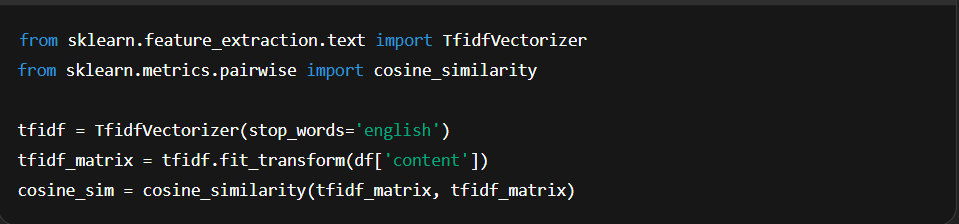
Where:

* TF(t, d) = Term frequency of term *t* in document *d*
* DF(t) = Number of documents containing term *t*
* N = Total number of documents
* Use **Cosine Similarity** to compute relevance between books:

**sim(A, B) = (A · B) / (||A|| ||B||)**

**3. Machine Learning Model**  
We use a **content-based filtering** approach with the following steps:

**Python code:**



**4. Deployment**

* **Deployment**: Interactive web interface via Streamlit
* **Hosting**: Locally or using Streamlit Cloud/Heroku
* **Evaluation**:
  + **Precision@5**: Are the top 5 recommendations actually relevant?
  + **Manual validation**: Check against real user interests
  + **User feedback**: Improve system over time with real usage

**SYSTEM APPROACH**

* **System Requirements:**Python 3.x, Jupyter Notebook, VS Code
* **Libraries Used:**
* pandas, numpy – Data handling
* **scikit-learn –** TF-IDF and similarity modelling
* **streamlit –** Web UI
* **Workflow:**Data Collection → Preprocessing → Feature Engineering → Similarity Modelling → UI Deployment

**ALGORITHM & DEPLOYMENT**

* **Algorithm Used**: Content-Based Filtering using TF-IDF & Cosine Similarity
* **Input Features**: Book metadata — title, author, genre, description
* **Training Process**:
  + Convert text to vectors using TF-IDF
  + Measure similarity using cosine distance
* **Prediction**:
  + For each book, find top 5 closest vectors
  + Return recommended books
* **Deployment**:
  + Web app via Streamlit
  + Fast, interactive, and scalable

**RESULT**

The system successfully recommends the top 5 books most similar to the one selected by the user. Results are displayed in real-time with relevant information like title, author, and genre. The output is quick, intuitive, and improves book discovery experience.

**CONCLUSION**

The Book Recommendation System using AI provides accurate and personalized book suggestions. It leverages content similarity to return relevant recommendations based on metadata. The system is scalable, user-friendly, and provides strong baseline performance with room for future improvement.

**FUTURE SCOPE**

* Add collaborative filtering using user ratings
* Incorporate NLP sentiment analysis on reviews
* Include multilingual support
* Enable real-time personalization
* Deploy on cloud platforms (AWS/GCP) with user login and feedback loop

**REFERENCES**

* **Kaggle Dataset:** Books Recommendation Dataset
* **GitHub Repository:** github.com/user/book-recommender
* **Paper:** *“*A Survey of Recommender Systems Based on Content Filtering”