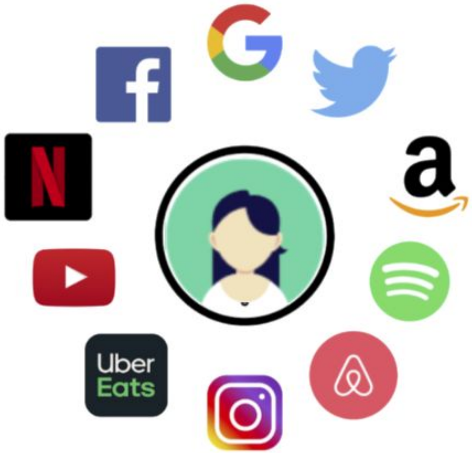
META RECOMMENDATION SYSTEM

- Chaitanya Attanti (23110052), Keerthan Varma (23110068), Avinash Reddy (23110123), Dinesh Siddhartha(23110168), Praneeth Pabbathi (23110226),

PROBLEM STATEMENT ?

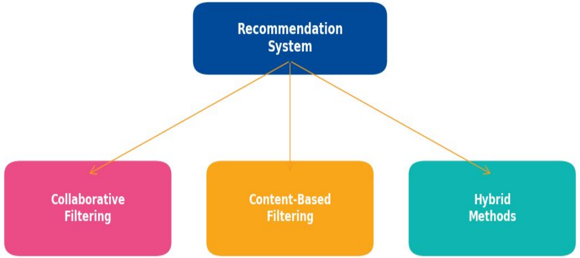
-Recommending the optimal recommendation system depends on the nature of the dataset, performance objectives, and specific project requirements. As different models excel under different conditions, there is no universal solution—choices must be guided by the data characteristics, user needs.



General Methods used in Recommendation System

1. Collaborative Filtering :

Recommends items based on **user-item interactions** like ratings or clicks.
Works by finding similar users or items, or using matrix factorization to learn hidden patterns.
Used by platforms like **Netflix**, **Amazon** to suggest personalized content.



2. Content-Based Filtering :

Recommends items that are **similar in content** to what the user has liked before.
Uses item features (genre, keywords, etc.) and builds a profile of user preferences.
Common in **job portals**, **news apps**, and **movie recommenders**.

3. Hybrid Methods :

Combine multiple approaches (e.g., collaborative + content-based) for better accuracy.
Can overcome limitations like cold start and provide more balanced recommendations.
Used in complex systems like **YouTube**, **Spotify** for highly personalized suggestions.

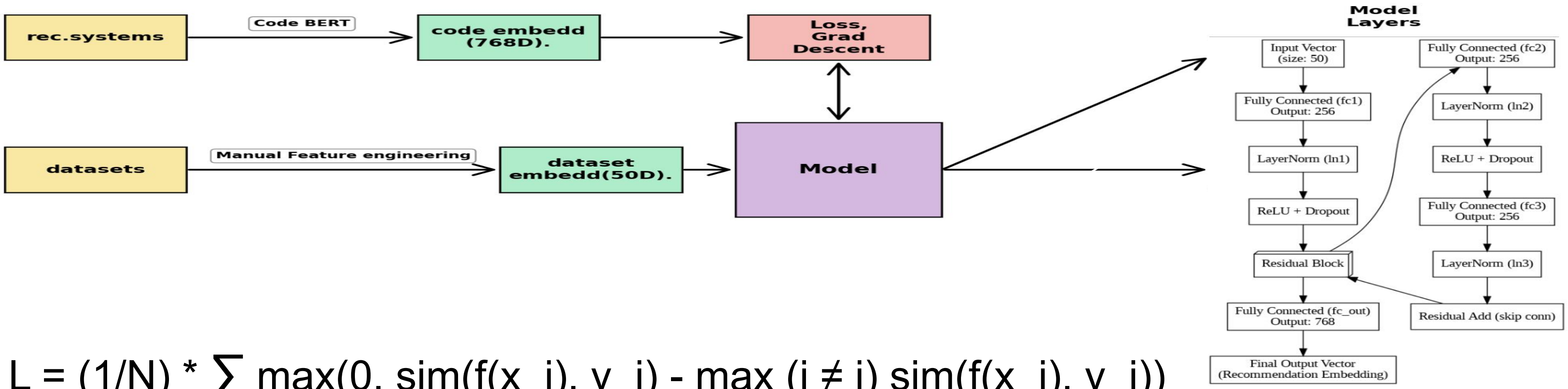
INPUT - OUTPUTS

Our system is designed to support all four input-output types:

- **Image-to-Image:** Recommending visually similar or related images (e.g., matching outfits).
- **Text-to-Image:** Retrieving relevant images from textual input (e.g., “red evening gow
- **Image-to-Text:** Generating captions or descriptive tags from images.
- **Text-to-Text:** Suggesting related content based on a given text (e.g., books or article

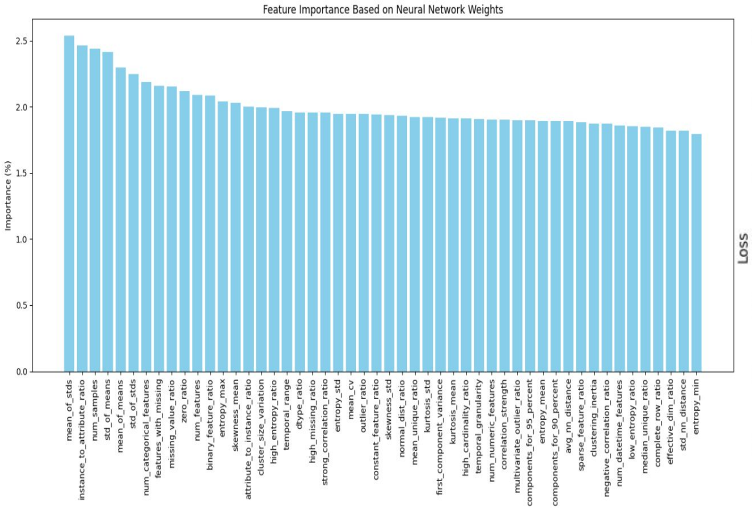
While we currently demonstrate only the **text-to-text** mode — due to the availability of open-source code, clean datasets, and ease of evaluation — our architecture is designed to be extensible to all the above input-output formats. With appropriate embeddings and modality-specific processing, the same core recommendation logic can be applied across image and text combinations.

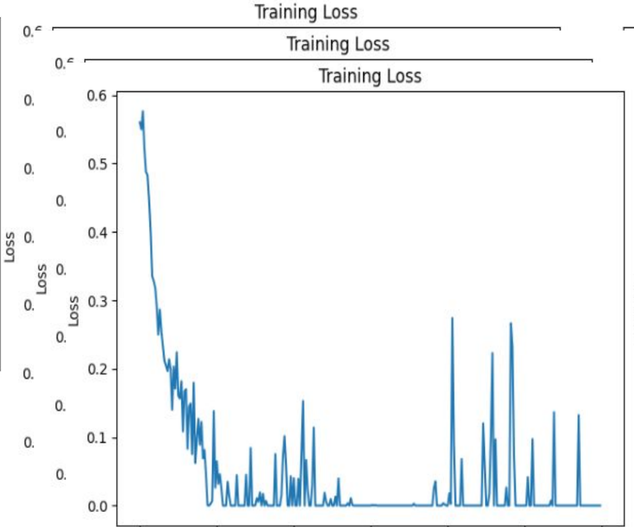
METHODOLOGY

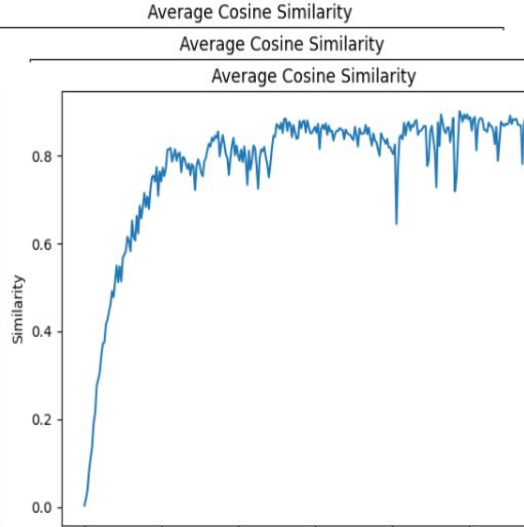


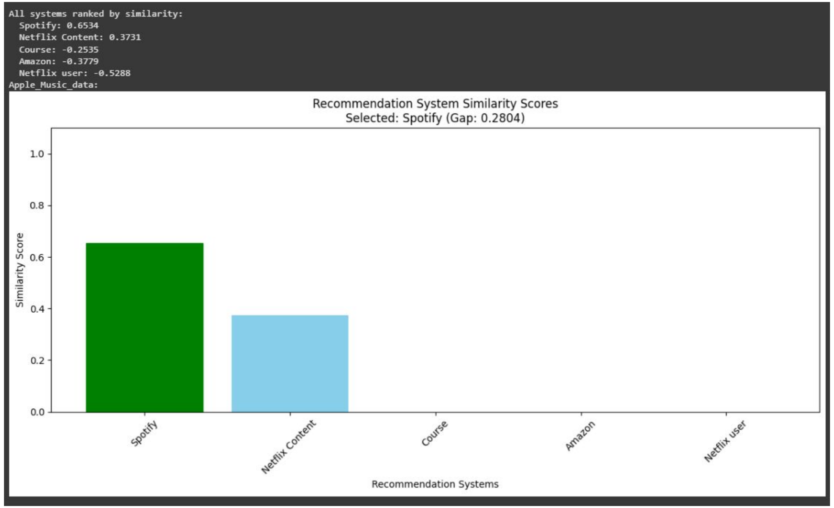
$$L = (1/N) * \sum \max(0, sim(f(x\_i), y\_i) - \max (j \neq i)\ sim(f(x\_i), y\_j))$$

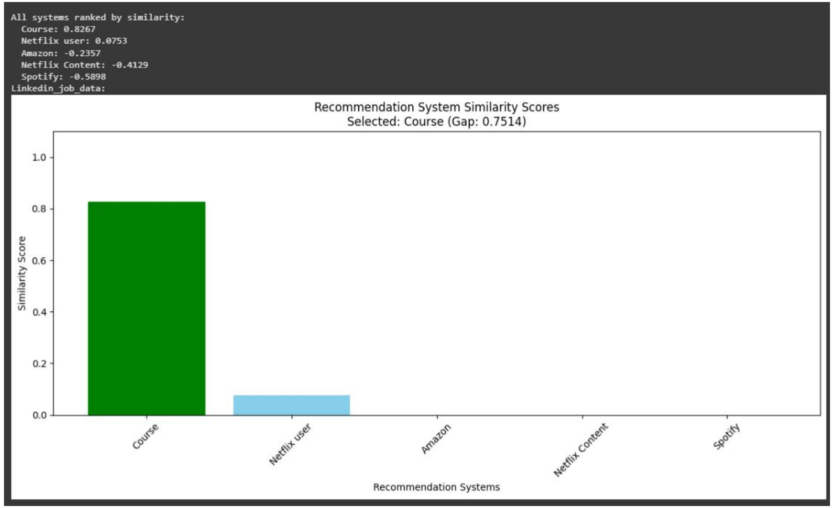
WORKING AND EXPERIMENTATION RESULTS

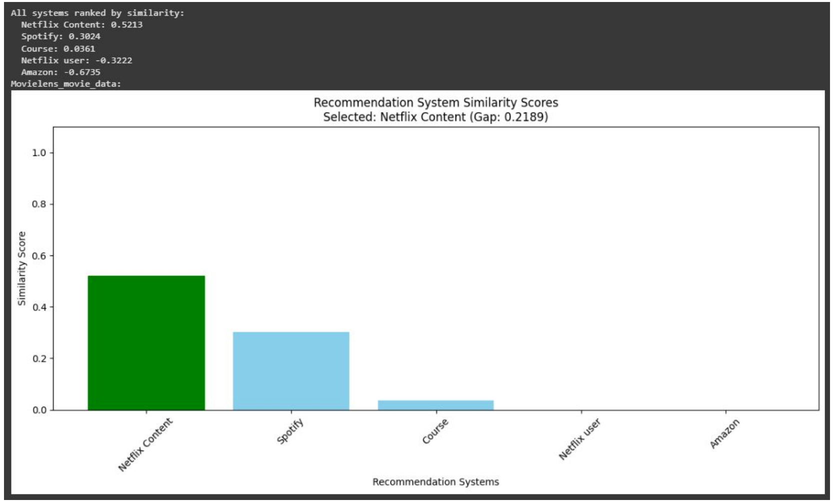


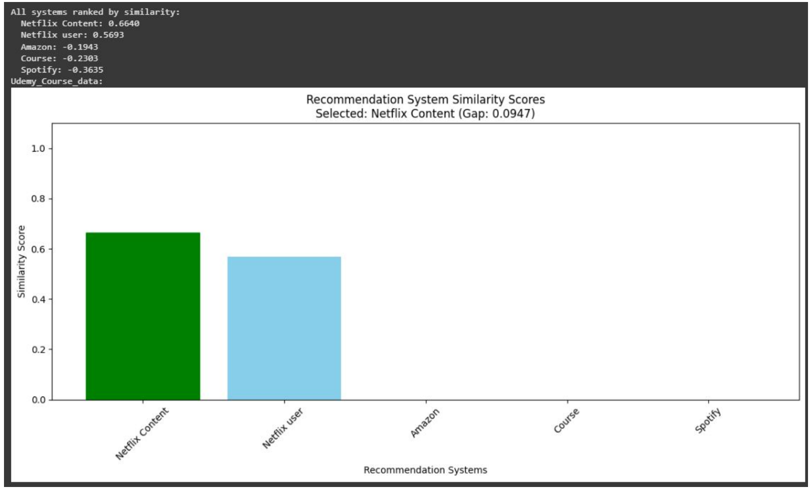





All systems ranked by similarity:
Spotify: 0.6534
Netflix Content: 0.3731
Course: -0.2535
Amazon: -0.3779
Netflix user: -0.5288
Apple_Music_data:
Recommendation System Similarity Scores
Selected: Spotify (Gap: 0.2804)
Similarity Score
Recommendation Systems
Spotify
Netflix Content
Course
Amazon
Netflix user

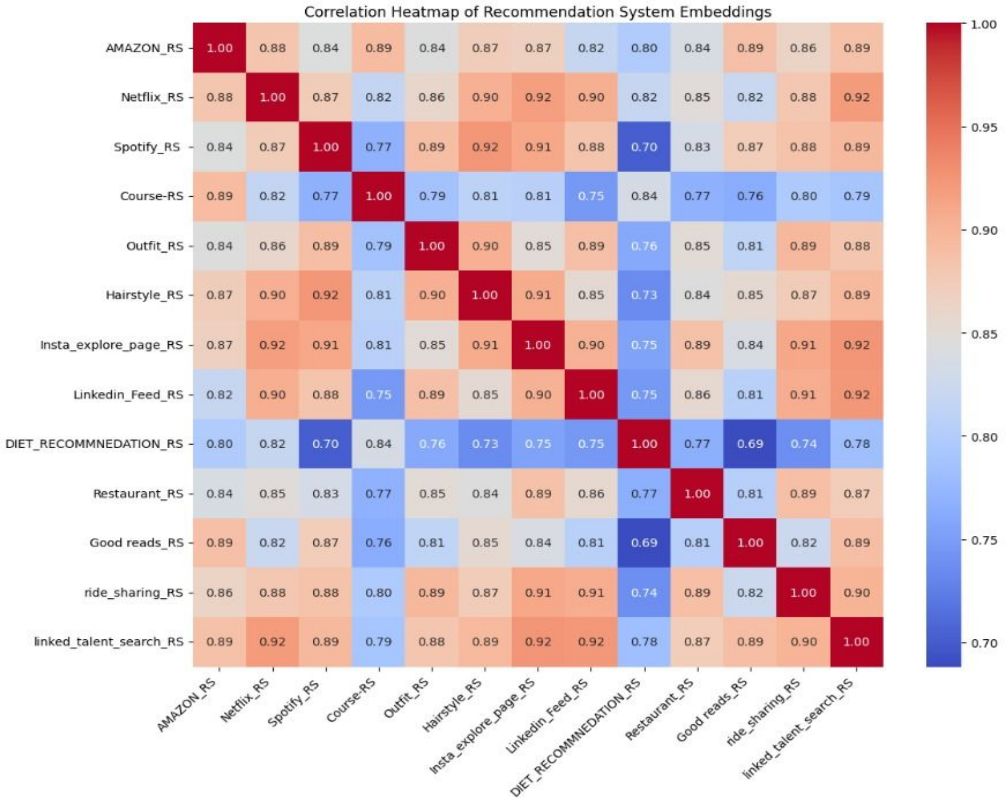

All systems ranked by similarity:
Course: 0.8267
Netflix user: 0.0753
Amazon: -0.2357
Netflix Content: -0.4129
Spotify: -0.5898
Linkedin_job_data:
Recommendation System Similarity Scores
Selected: Course (Gap: 0.7514)
Similarity Score
Recommendation Systems
Course
Netflix user
Amazon
Netflix Content
Spotify


All systems ranked by similarity:
Netflix Content: 0.5213
Spotify: 0.3024
Course: 0.0361
Netflix user: -0.3222
Amazon: -0.6735
Movielens_movie_data:
Recommendation System Similarity Scores
Selected: Netflix Content (Gap: 0.2189)
Similarity Score
Recommendation Systems
Netflix Content
Spotify
Course
Netflix user
Amazon


All systems ranked by similarity:
Netflix Content: 0.6640
Netflix user: 0.5693
Amazon: -0.1943
Course: -0.2303
Spotify: -0.3635
Udemy_Course_data:
Recommendation System Similarity Scores
Selected: Netflix Content (Gap: 0.0947)
Similarity Score
Recommendation Systems
Netflix Content
Netflix user
Amazon
Course
Spotify

IS USING BERT MODEL FOR EMBEDDING RIGHT ?

The bert model was able to get good correlations between recommendation just by using text data of each system
Example:
Hairstyle Style and outfit style which have almost same process for recommendation got high similarity score, diet recommendation system which has input and output different which is very unusual compared to other recommendation system got a very bad similarity score compared to all other recommendation system .

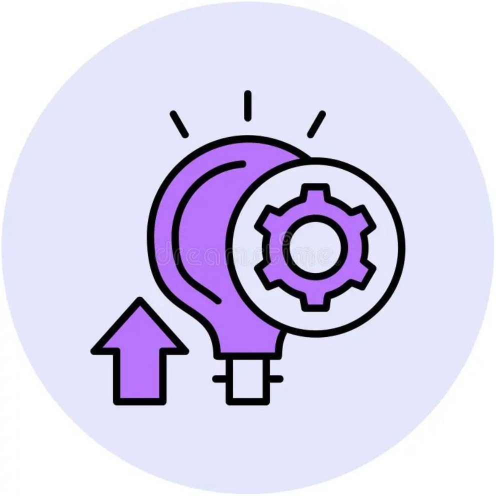


NOVELTY

We built a **Meta-Recommender** — a system that recommends the best *recommendation system* for a particular dataset and use case.

What's Novel?

- **Reverse Recommendation:** Recommending the right recommender system based on the problem and data.
- **Cross-Modality Matching:** Matching the meaning of the model's code with the features of the dataset, even though they come from very different formats (text,numbers) — something not commonly done in recommendation system selection.
- **Evaluation Metric:** Δ = Correct System Similarity − Max Incorrect Similarity as a robustness measure.



LIMITATIONS

- **Data Dependency :**
  - Performance heavily depends on the quality and quantity of available data.
  - Sparse or biased datasets can lead to unreliable model evaluation.
  - Also, The need of user could be something, which is not taken into account, but can be included.
- **Domain Transferability :**
  - A model or metric that works well in one domain (e.g., movies) may not perform equally well in another (e.g., e-commerce) without adjustments.

FUTURE WORK

- **Multi-Model Support:**
  - Expand the system to handle not just text, but also images and other data types.
  - Enable recommendations across different formats (e.g., suggest images based on text, or vice versa).
- **Automated Feature Extraction:**
  - Automate the process of extracting important features in data.
  - This reduces the need for manual work and makes the system more scalable and adaptable.
- **Enhanced Model:**
  - Ensuring recommendations take into account, the need of the user.

CONCLUSION

- Novel Approach: Successfully created a meta-recommender that suggests appropriate recommendation systems based on dataset characteristics and requirements
- Dual-Feature Architecture: Combined BERT embeddings with hand-crafted features to capture both semantic relationships and domain-specific attributes
- Proof of Concept: Demonstrated viable results with just 5 base recommendation models, establishing foundation for more comprehensive meta-recommendation systems
- Future Potential: Framework shows promise for expansion to multi-modal data types and integration of additional recommendation algorithms

EXTRA FINDINGS

- We experimented to find a proxy to cosine similarity function-

Metric Comparison Across Recommendation Systems

Metric	Netflix Overlap	Netflix Spearmann Corr	Spotify Overlap	Spotify Spearmann Corr	Amazon Overlap	Amazon Spearmann Corr
Sigmoid Dot / Proj	52/100	0.05	95/100	0.951	40/250	-0.152
ReLU-Cosine	100/100	1.0	100/100	1.0	100/250	1.0
Tanh-Dot	39/100	-0.12	97/100	0.803	40/250	-0.152
Inverse Hyperbolic	100/100	0.86	100/100	0.97	240/250	0.94
Norm-Contrast	100/100	0.76	99/100	0.975	46/250	0.18

We find that the Inverse Hyperbolic Tanh similarity closely mimics cosine similarity across datasets, and can serve as a reliable proxy.

$$\mathrm{IHT}(x,y)=1-\tanh(\arccos(\mathrm{cosine\_sim}(x,y)))$$