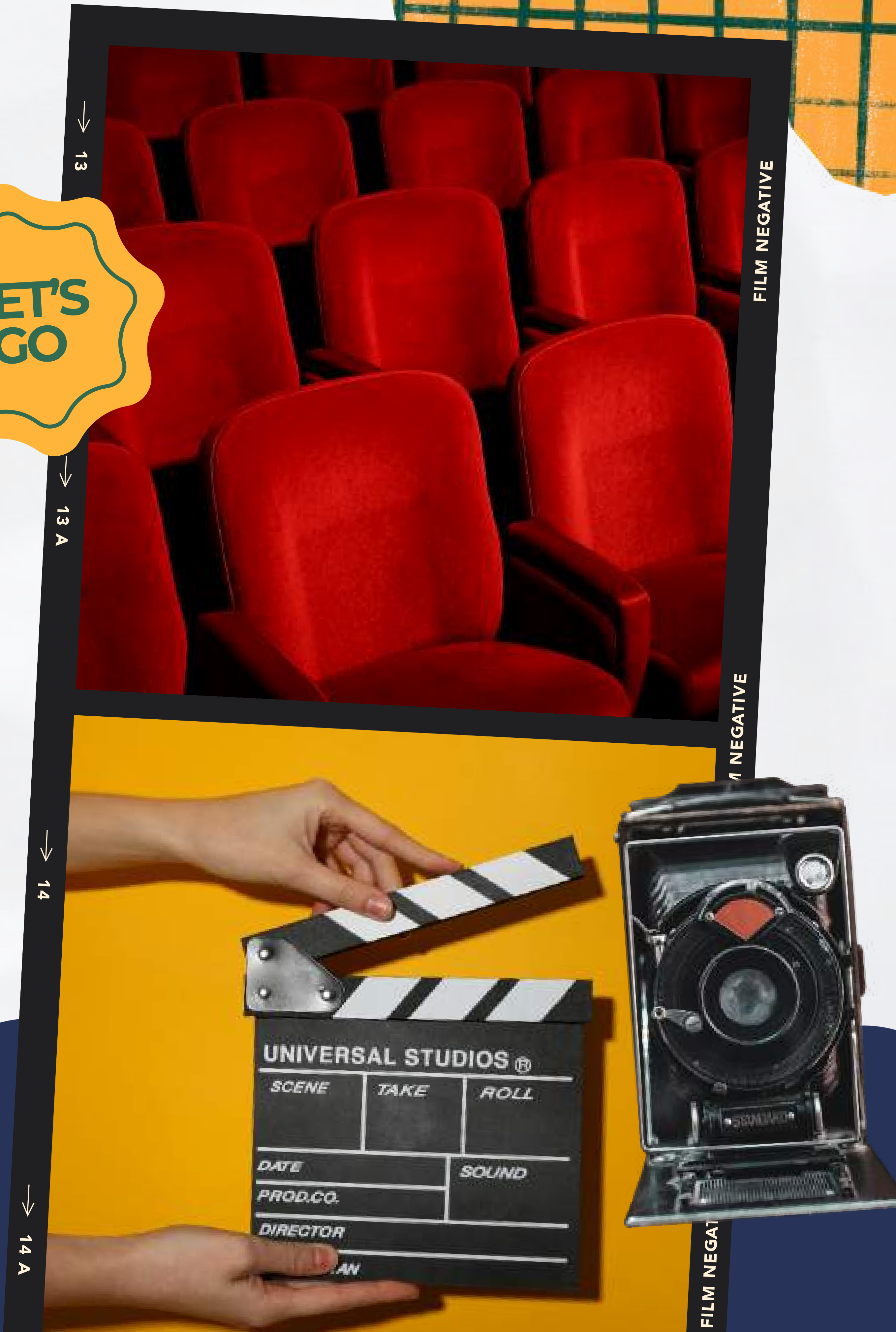
ARCHIVER

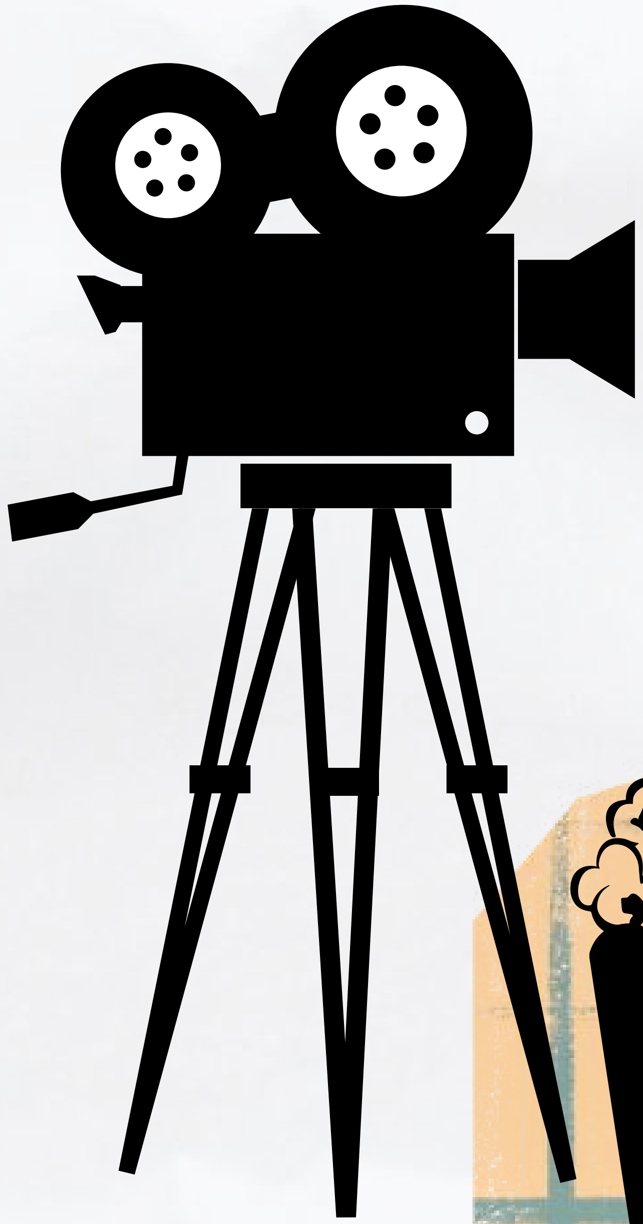
LET'S GO

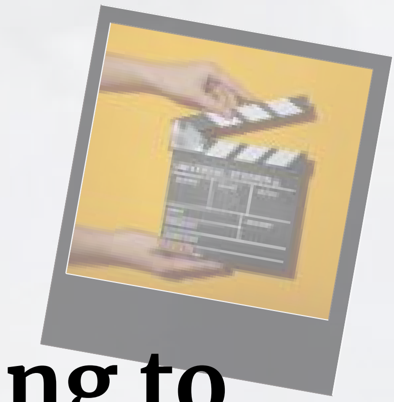
A Hybrid Movie Recommendation System



Problem Statement

- Users Face Information Overload due to millions of movies online
- Steaming platforms promote their own content, not necessarily what serves the user's interests
- Existing Systems (Content-based, Collaboration) face :
    1. Cold-start
    2. Data Sparsity
    3. Population bias
- Platforms often show biased, promotional content, leading to poor Personalization



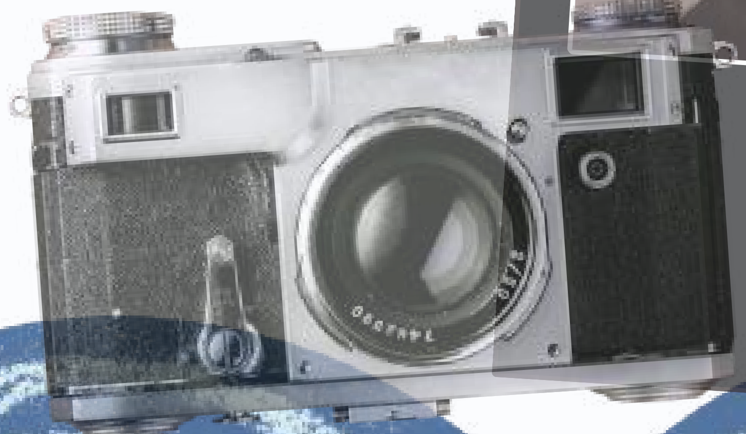


Motivation

- Need for an independent, unbiased recommendation platform
- Deliver personalised suggestions based on :
  - User Preferences
  - Genuine Content similarity
  - Viewing Patterns
- Goal: Improve accuracy, adaptability and user Satisfaction

Key Challenges

- Information overload
- Cold-start for new users/movie
- Sparse rating matrices
- Scalability
- Limited personalization in existing mode



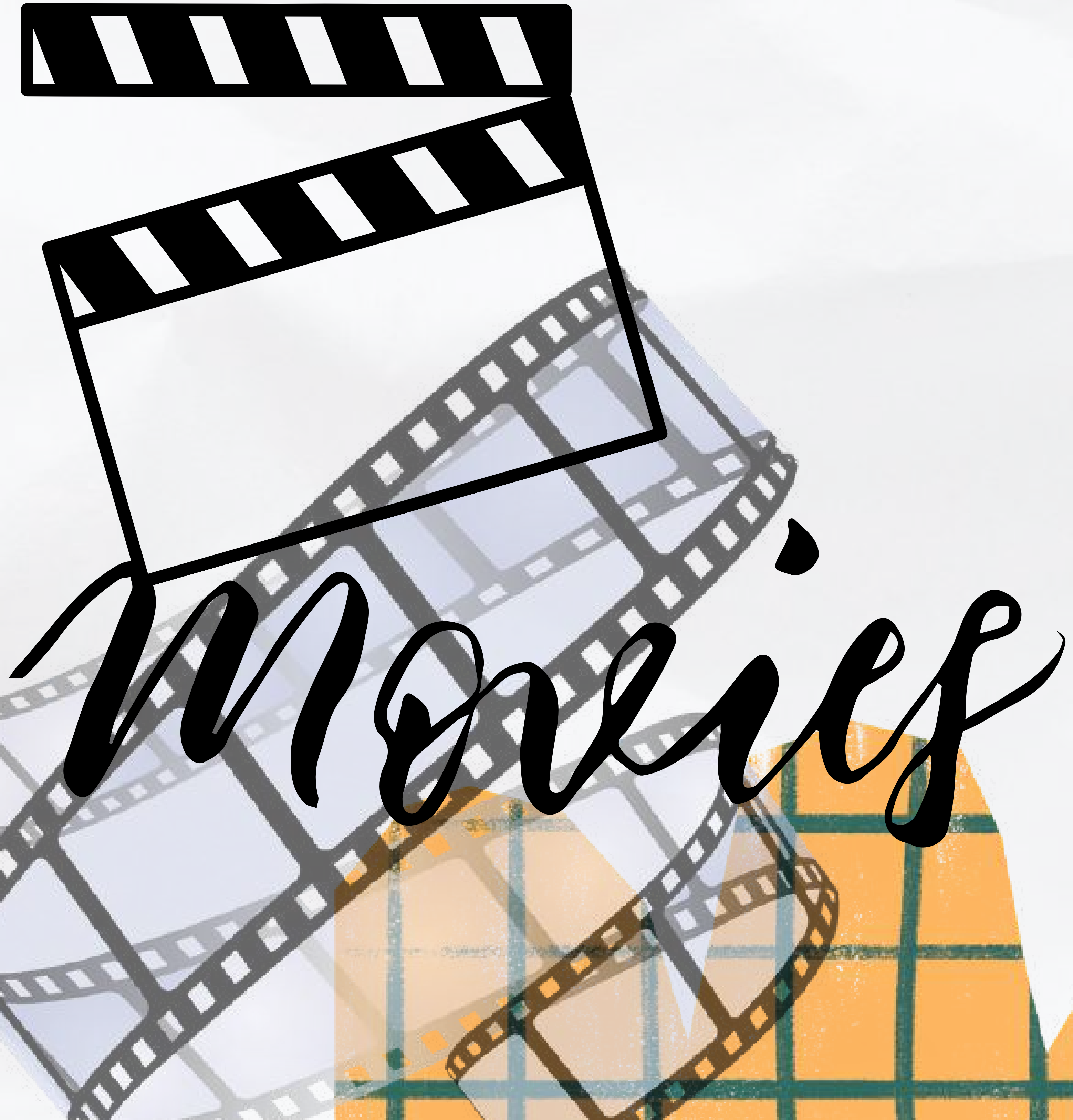
Proposed Hybrid Approch

The model integrates four ML techniques :

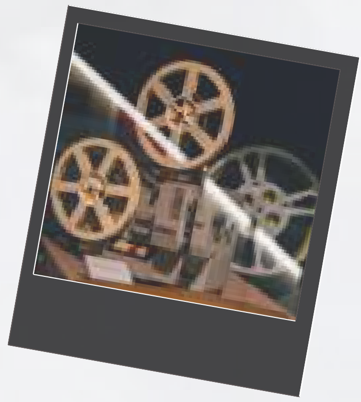
- K-Means : User & Movie Clustering
- Linear Regression : Rating Prediction
- Apriori: Association rule mining
- Cosine Similarity : item similarity

Combines using hybrid scoring for final recommendation

Dataset Details



- IMDb Indian Movies Dataset (~6000 records)
- Includes:Titles, genres, cast, tags, Ratings, metadata
- Balanced mix of numerical + textual features.



k-Means Clustering

Purpose : Clusters movies into natural groups

Steps:

- Euclidean distance as metric
- Optimal k using Elbow method

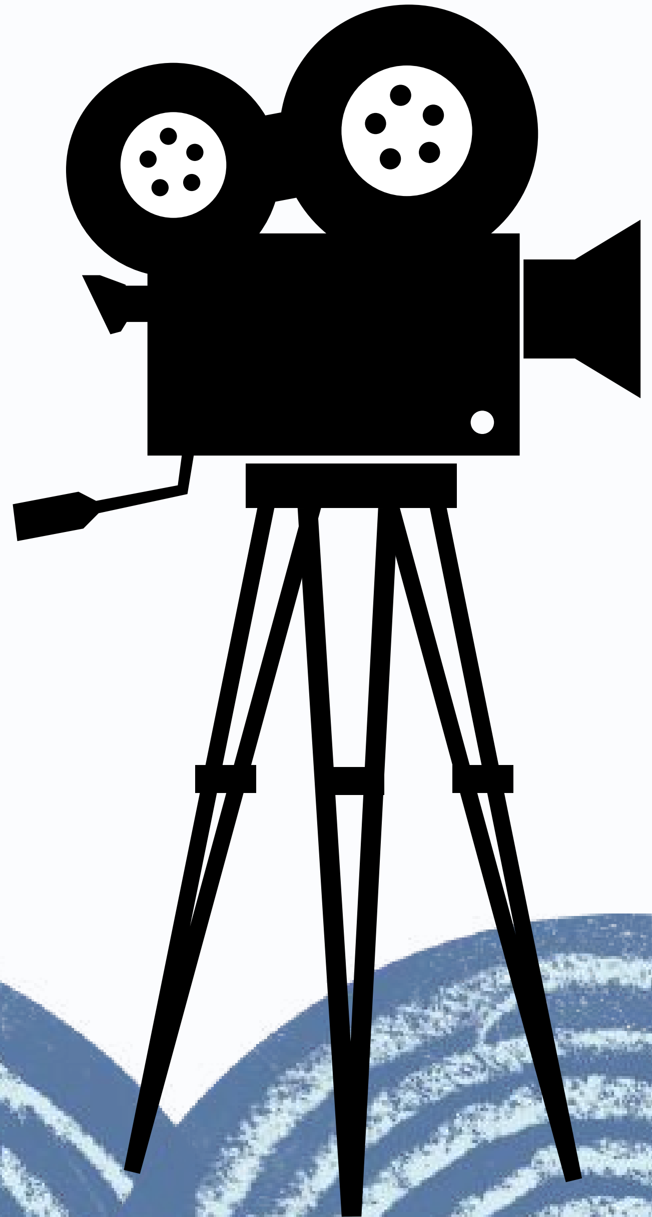
Output:

- Cluster assignments
- Cluster centroids → dominant preference patterns

Benefits :

- Faster similarity search
- Better cold-start handling

Evaluation metrics :



Linear Regression

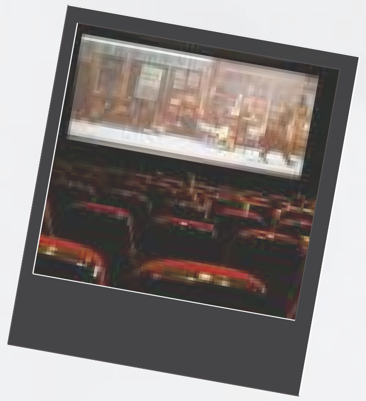
Goal : Predict user rating for unseen movies
Model : $y=\beta0+\beta1x1+...+\beta nxn$
Process :

- 80% training, 20% testing
- Gradient-based optimization

Output :

- Predicted ratings - ranking factor in recommendations

Evaluation Metric :



Apriori Algorithm

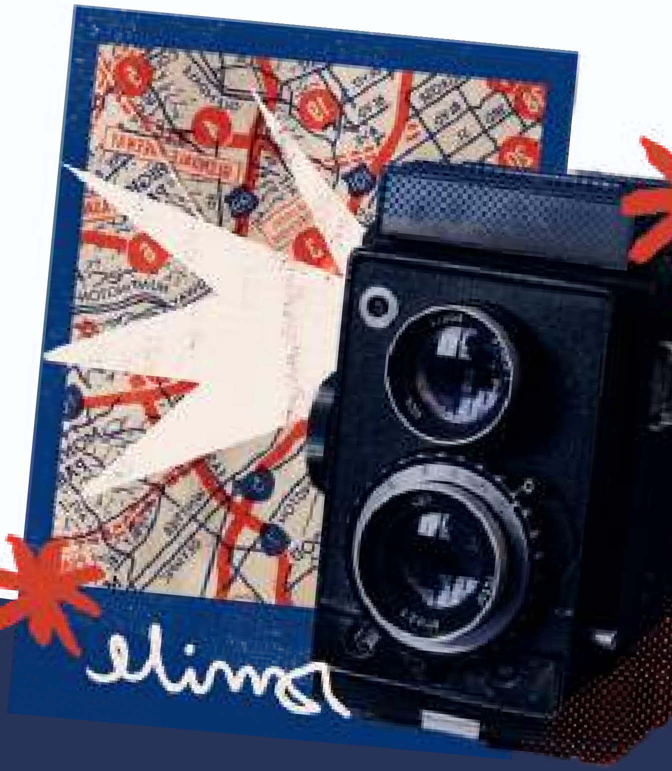
Goal: Discover viewing patterns & associations

Key Metrics :

- **Support :** Filters out rare or irrelevant combinations and keeps only popular patterns.
- **Confidence :** Helps measure strength and reliability of the association.
- **Lift :** Lift helps ensure recommendations are not coincidental but genuinely related.

Outputs:

- Rules Like: User who watched A also watched B
- Pattern driven recommendations



Cosine Similarity

Goal: Compute movie-movie similarity

Formula:

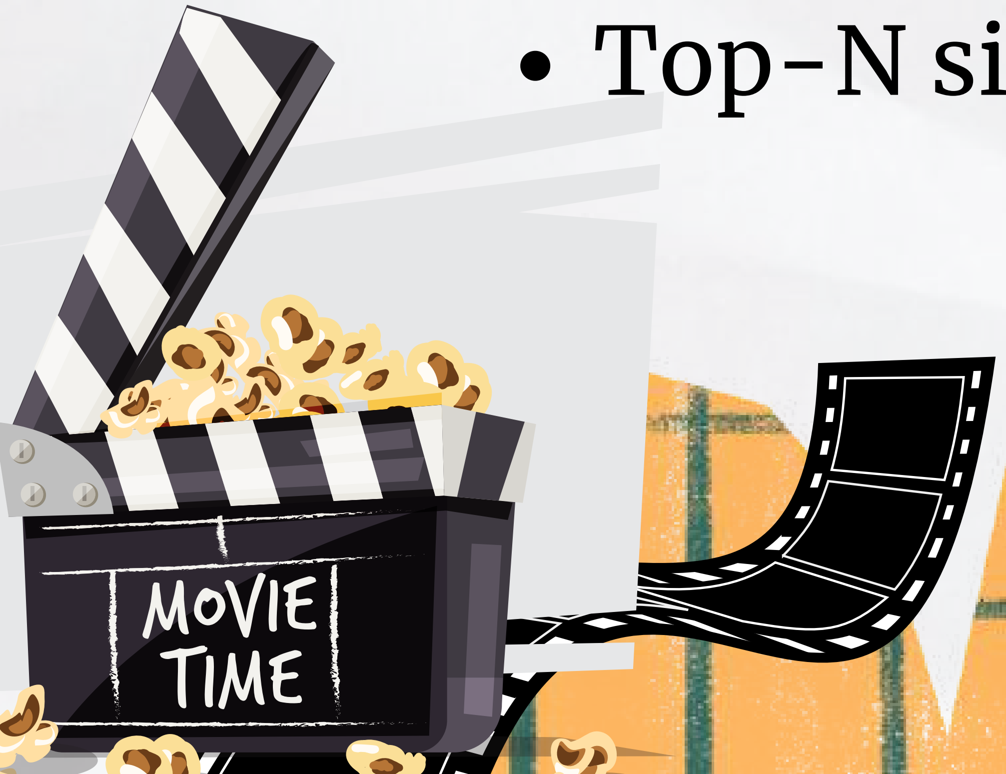
$\text{CosSim}(A,B)=A \cdot B\ /(\,|\,A\,|\,)(\,|\,B\,|\,)$

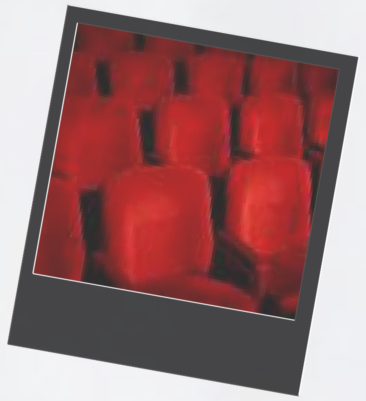
Why Effective :

- Works well on sparse vectors
- Genre/tag based similarity

Output :

- Top-N similar movies





Evaluation Metrics

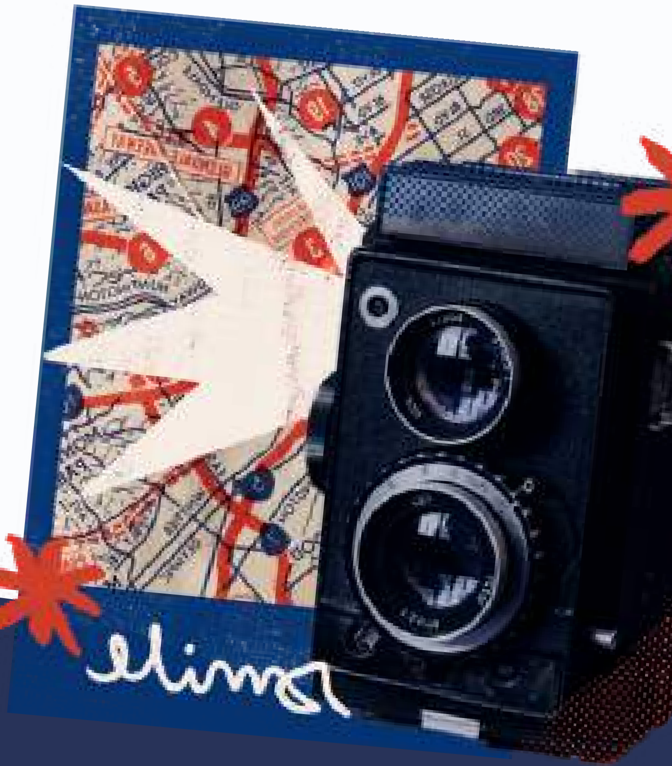
Cluster distribution:

Cluster	
0	259
1	20
2	90
3	19
4	2728
5	847
6	1237
7	405
8	34
9	20

Performance evaluation of Random Forest Regressor:
Mean squared error: 1.256169237466984
Mean absolute error: 0.9523022086875719
R2 score: 0.17871139093747979

Performance evaluation of Decision Tree Regressor :
Mean squared error: 0.09026800670016752
Mean absolute error: 0.08877721943048589
R2 score: 0.9526017807828623

Performance evaluation of Linear Regression:
Mean squared error: 0.41726604724835503
Mean absolute error: 0.48133520277992636
R2 score: 0.7809005836914157



experience

THANK You