Movie Recommendation System

Wrik Bhadra
MT18027

Gurpreet Singh
MT18098

Dhruv Kaushik
MT18023

IIIT Delhi, New Delhi 110020, India

Problem Statement

Recommender systems have become ubiquitous in our lives. Be it e-commerce websites or social media platforms, recommender systems add the "what-next" factor to it. Due to the advances in recommender systems, users constantly expect good recommendations. They have a low threshold for services that are not able to make appropriate suggestions. This has led to a high emphasis by tech companies on improving their recommendation systems. However, the problem is more complex than it seems.

In this project, we aim to give personalized recommendations to users based on the movies that they have already rated.

Literature Review

Content-based filtering makes recommendation based on similarity in item features. Popular techniques in content-based filtering include the term frequency / inverse document frequency (tf-idf) weighting technique in information retrieval [1][2] and word2vec in natural language processing. An extension of word2vec, called doc2vec [3] is used to extract information contained in the context of movie descriptions.

Content-based filtering works well when there hasn't been enough users or when the contents haven't been rated. Collaborative filtering recommends items that similar users like, and avoids the need to collect data on each item by utilizing the underlying structure of users' preference. One major approach in collaborative filtering is neighborhood model [4]. The neighborhood model recommends the closest items or the closest user's top-rated items.

Dataset Description

Dataset is provided at Kaggle as The MovieLens Dataset [4]. The original dataset contains data of 45,000 movies with features like cast, genre, revenue, language, release date, etc. The whole dataset contains 26 million ratings rated by 270,000 users. A rating is a decimal value between 0 and 5 in multiples of 0.5.

Our project considers a subset of the original data comprising of details of 4320 movies and a total of 1,19,000 ratings given by 6000 users.

Method 1 : Item-based collaborative filtering

A. Phase 1: Baseline model

- Total 6 features are selected for representing a movie. They are divided into two feature space:
  - F1 = {Actors, Directors, Genres, Country, Keywords}
  - F2 = {Overview}
- Number of tokens in Overview feature is usually as large as the total number of tokens in remaining features combined. Thus, to prevent underweight of F1 features, the 6 features are broken in F1 and F2 feature spaces.
- For each movie, two tf-idf feature vectors are obtained, one corresponding to F1 and another corresponding to F2.
- For each vector space, we generate a similarity matrix using cosine similarity.
- The overall similarity score, sim(i,j) between movie i and movie j is obtained as :

$$\mathrm{sim}(i,j) = \mathrm{alpha} * \mathrm{sim1}(i,j) + (1 - \mathrm{alpha}) * \mathrm{sim2}(i,j)$$

simδ(i, j): value of cell (i, j) in Similarity Matrix δ
alpha: arbitrary weight in the range [0,1]

Rating predicted of movie i for user u

$$\widehat{r_{ui}} = \mu_u + \frac{\sum_j sim(i,j)(r_j - \mu_u)}{\sum_j sim(i,j)}$$

where μ_u is the average rating done by user 'u'.
r_j is rating of movie 'j'.
$\widehat{r_{ui}}$ is predicted rating of user 'u' for movie 'i'.
$sim(i,j)$ is similarity value between movie 'i' and movie 'j'.

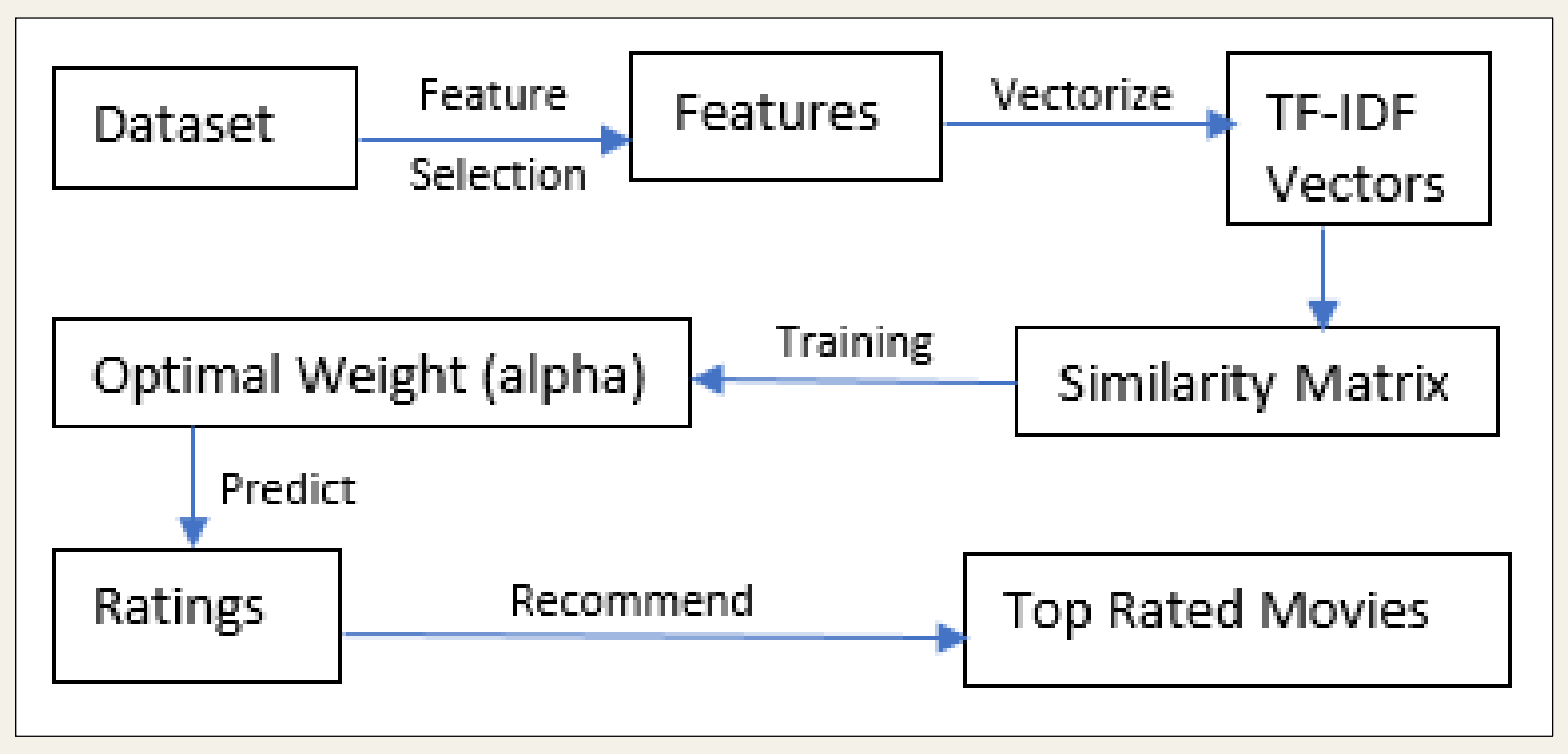

Fig. 1: Pipeline followed in Phase 1

B. Phase 2: Feature Selection

- Forward sequential Wrapper method applied over F1 features to get best combinations of F1 features.

C. Phase 3: Dimensionality reduction

- Top 5 feature combinations are selected from Phase 2 based on Test RMSE.
- PCA over the 5 feature combinations to get their performance in lower dimension.

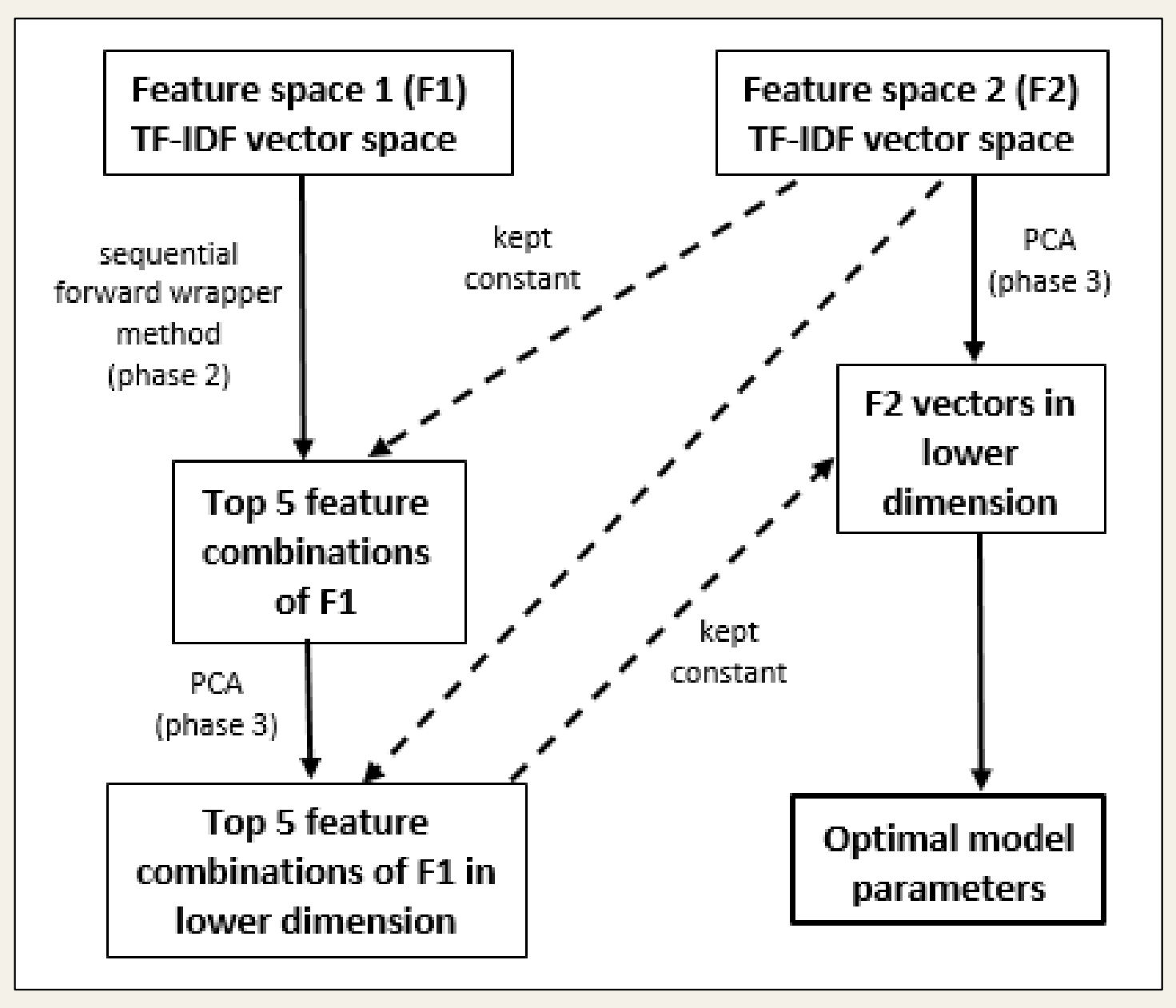
Feature space 1 (F1) TF-IDF vector space
Feature space 2 (F2) TF-IDF vector space
sequential forward wrapper method (phase 2)
kept constant
PCA (phase 3)
F2 vectors in lower dimension
Top 5 feature combinations of F1
PCA (phase 3)
kept constant
Top 5 feature combinations of F1 in lower dimension
Optimal model parameters

Fig. 2 : Pipeline followed in Phase 2 and 3

Method 2 : k-Nearest Neighbor (kNN)

- TF-IDF based vectors obtained for all movies over all 6 features combined (single feature combination).
- For each user, 80 % of his ratings are taken as training set, ratings for remaining 20 % are predicted. (test set)
- PCA is performed for different values of eigen-energy varying from 50% to 99% (with an increment of 5%).
- kNN is run, for k = 1 to 4, individually for each eigen-energy.

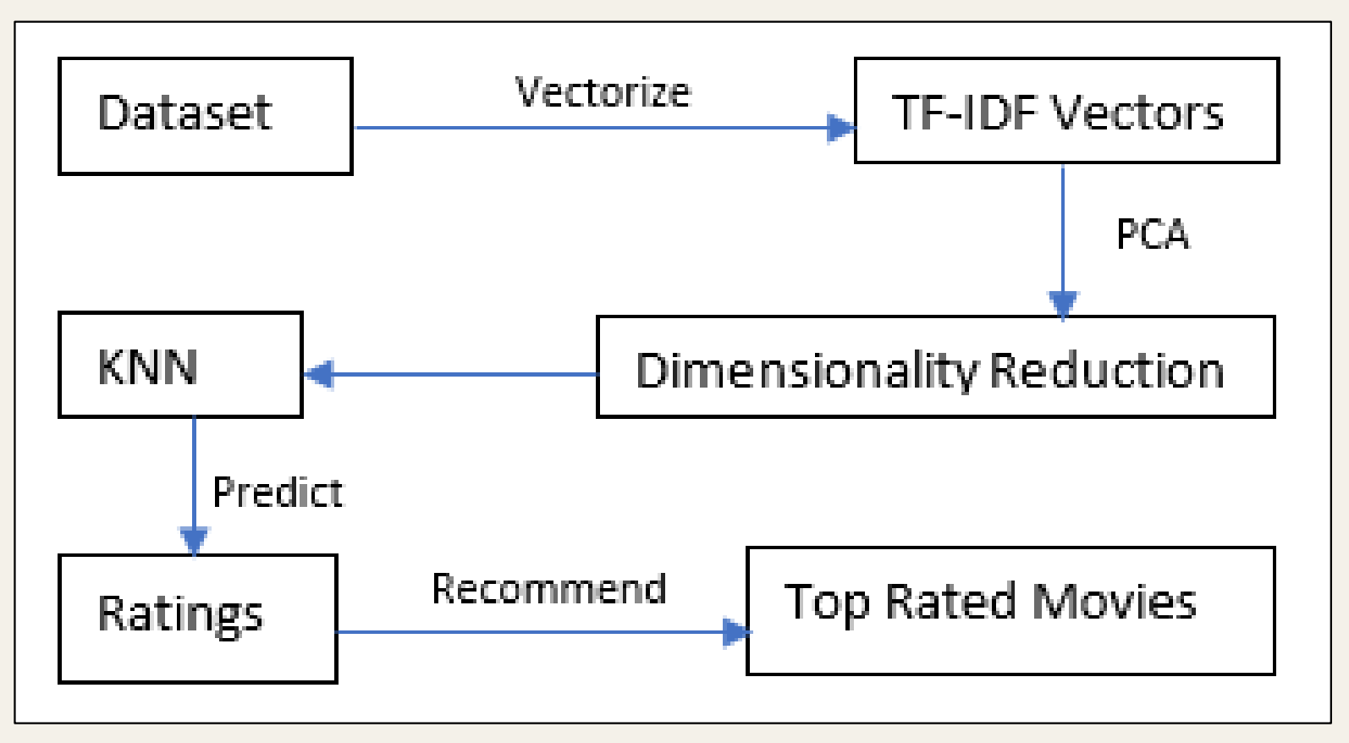

Fig. 3 : Pipeline followed in Method 2

Tools and Technologies

- Language: Python
- Libraries
  - NLTK
  - Scikit-learn
  - SciPy
  - Matplotlib

Evaluation metric

RMSE score is calculated as follows:

$$RMSE = \sqrt{\sum_{i=1}^{N} (pred_i - true_i)^2 / N}$$

where,
$pred_i$: predicted value of the ith sample
$true_i$: true value of the ith sample
N : total number of samples

Results

A. Phase 1

- For alpha = 0.25, we got minimum RMSE value = 0.9611 as shown above.
- For this value of alpha, we found the test set RMSE value = 0.9409.

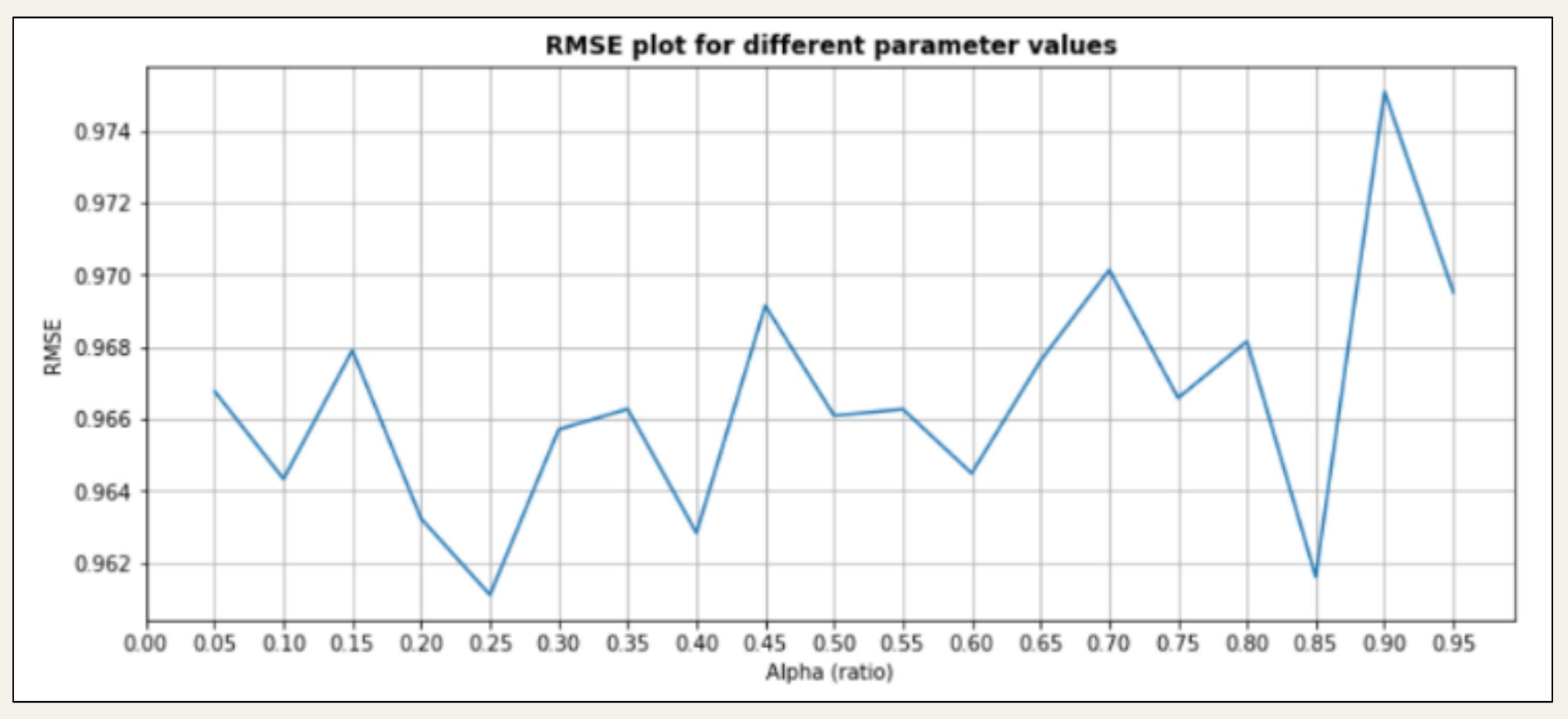

Fig. 4: Plot of RMSE over test set for different values of alpha

B. Phase 2

F1 Features	Alpha	Train RMSE	Test RMSE
Actor	0.90	0.9622	0.9531
Director	0.45	0.9609	0.9342
Keyterms	0.20	0.9663	0.9542
Genre	0.15	0.9623	0.9408
Country	0.15	0.9654	0.9307
Country, Actor	0.30	0.9637	0.9486
Country, Director	0.30	0.9655	0.9501
Country, Keyterms	0.30	0.9636	0.9529
Country, Genre	0.40	0.9633	0.9346
Country, Genre, Actor	0.30	0.9635	0.9437
Country, Genre, Director	0.70	0.9643	0.9536
Country, Genre, Keyterms	0.30	0.9623	0.9542
Country, Genre, Actor, Director	0.20	0.9653	0.9400
Country, Genre, Actor, Keyterms	0.60	0.9601	0.9384
Country, Genre, Actor, Keyterms, Director	0.60	0.9628	0.9476

Table 1: Results for forward sequential wrapper method

- Best feature combination for F1: {Country, Genre, Actor, Keyterms}
- Best value of alpha for this feature combination is 0.60

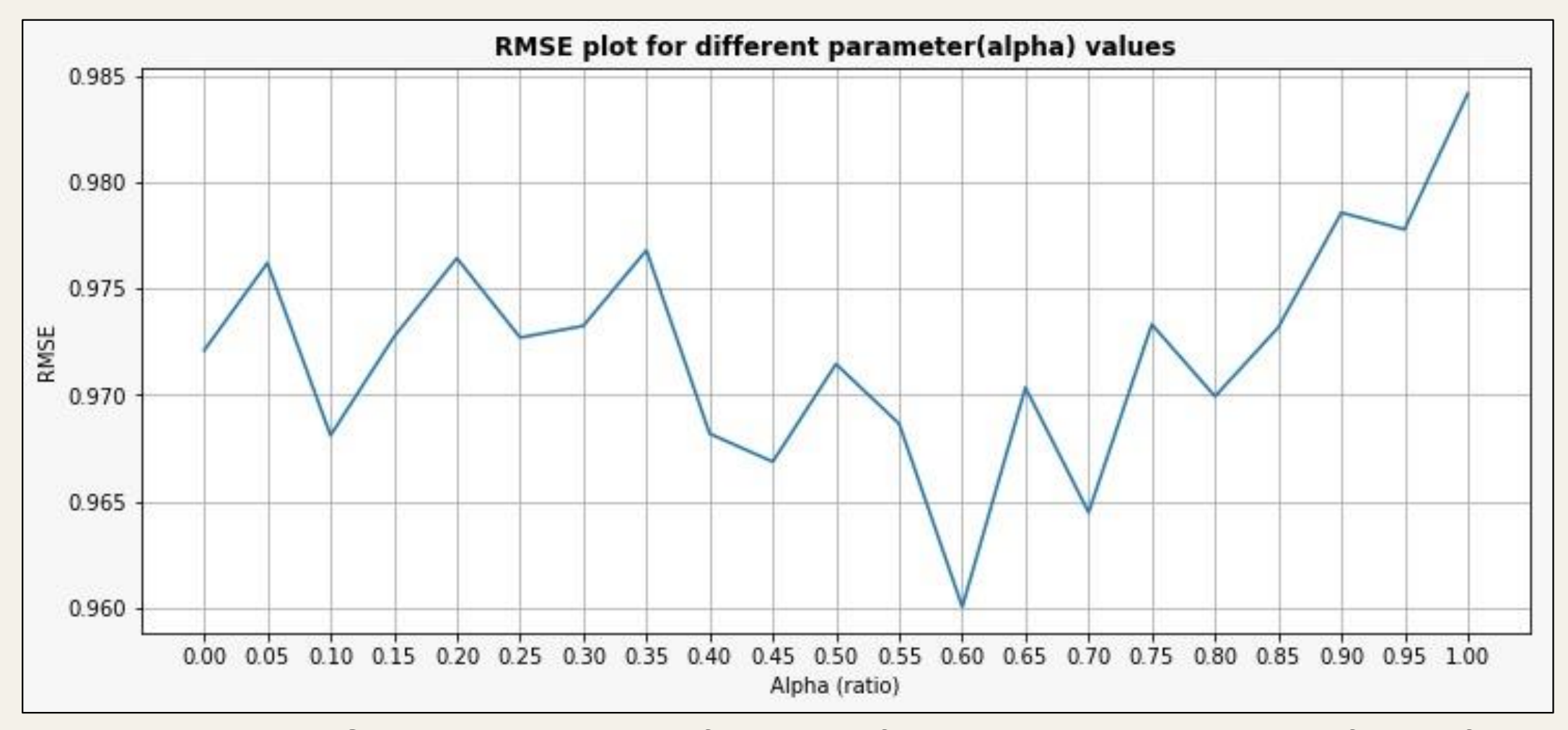

Fig. 5: Test RMSE vs alpha plot for best feature combination of F1 after wrapper method

C. Phase 3

F1 Features	Variance Ratio	F1 Dim	F2 Dim	Alpha	Test RMSE
Country	0.70	5	22180	0.40	0.9241
Country, Genre	0.99	71	22180	0.35	0.9345
Country, Genre, Actor	0.90	3328	22180	0.60	0.9308
Country, Genre, Actor, Director	0.60	1817	22180	0.70	0.9280
Country, Genre, Actor, Keyterms	0.60	1623	22180	0.70	0.9276

Table 2: Results of PCA over selected feature combinations

- Best feature combination for F1 after PCA: {Country}
- Best value of alpha for this feature combination is 0.40

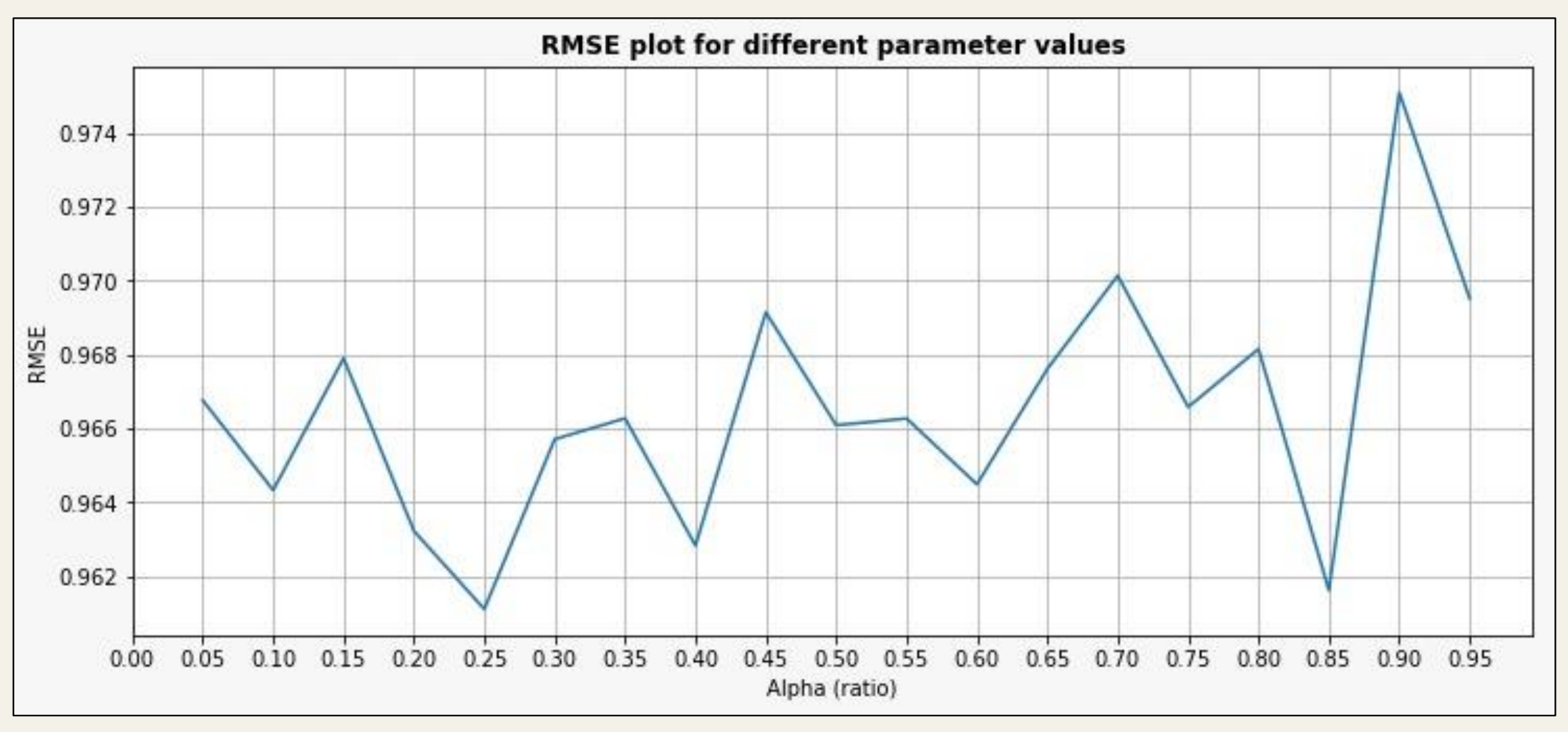

Fig. 6: Test RMSE vs alpha plot for best feature combination of F1 after PCA

Method 2

K	Feature Dim	Test RMSE
1	52257	1.3336
2	52257	1.3780
3	52257	1.4170
4	52257	1.3586

Table 3: Results of kNN without dimensionality reduction

- kNN works best for k = 1 in higher as well as lower dimension.
- K is limited in [1, 4] as each user in our dataset has rated minimum 5 movies

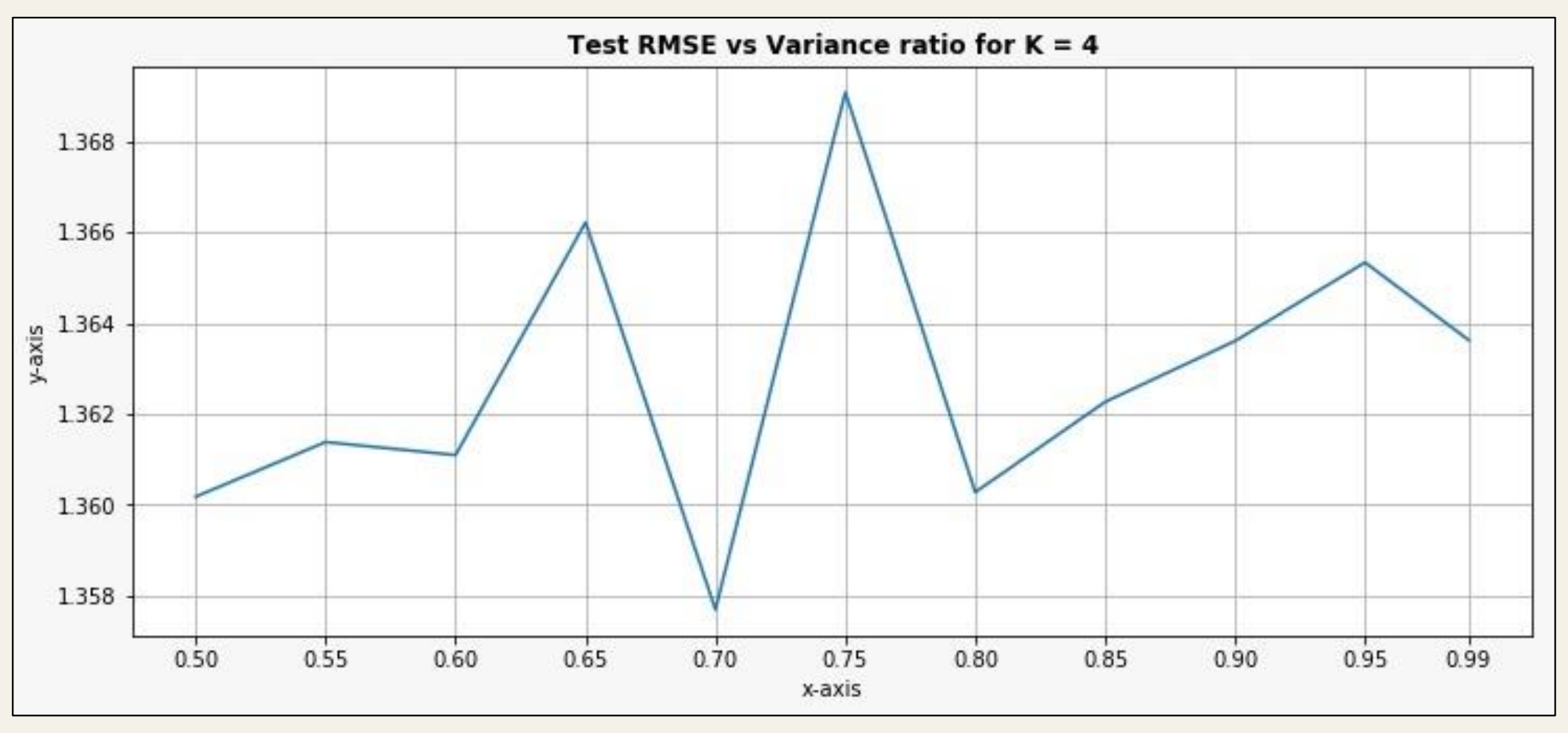

Fig. 7: Plot of test RMSE vs variance ratio in kNN for k = 4

Conclusion

- Country plays the most significant role among all the 6 features in rating a movie.
- Genre and actor taken together play the second-most significant role.
- In method 1, PCA helps in reducing test RMSE slightly (by around 0.01).
- For higher as well as lower dimension of feature vectors, we observe that k = 1 gives best and k = 4 gives second-best performance in kNN.

References

[1] A. Tuzhilin and G. Adomavicius. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge & Data Engineering, vol. 17, no.6, pp. 734-749, 2005.
[2] G. Salton. Automatic Text Processing. Addison-Wesley (1989)
[3] models.doc2vec - Doc2vec paragraph embeddings. Internet: https://radimrehurek.com/gensim/models/doc2vec.html, April 2009 [April 30th, 2019]
[3] The Movies Dataset. Internet: https://www.kaggle.com/rounakbanik/the-movies-dataset, April 2018 [March 2019]

Team members' contributions

Gurpreet Singh (MT18098) performed item-based collaborative filtering on feature vectors of high dimension.
Dhruv Kaushik (MT18037) performed feature selection using wrapper method and dimensionality reduction (PCA) to get best feature combinations.
Wrik Bhadra (MT18027) performed kNN on all feature vectors with and without dimensionality reduction.