Universidad Politécnica
de Madrid

**Escuela Técnica Superior de
Ingenieros Informáticos**

Master in Data Science

Information Retrieval: Comparison and Discussion of Two Profile-based Retrieval Systems

Rodrigo Castañón Martínez
Javier Arteaga Puell
Dakota James Mellish

In recent years, recommendation systems have grown more and more complex due to advancements in AI and technology. Major platforms have emerged such as Spotify, Netflix and Amazon and each provides tailored recommendations to users based on such systems. General approaches for providing recommendations include user-based, item-based and hybrid content filtering. We provide an overview of each method and analyze a recent 2016 hybrid-filter system proposal [1] provided by researchers Nitin Pradeep Kumar and Zhenzhen Fan from the University of Singapore.

To begin, it is worth explaining what each method does. User Based Collaborative Filtering and Item-Based Collaborative Filtering have been around since the late 90s [2] and early 2000s [3]. They were introduced by researchers at the University of Wisconsin and University of Hong Kong, respectively. They both involve the use of metrics such as cosine similarity, pearson correlation and Jaccard coefficient, either between users' or items' attributes. [4]. The challenge of these approaches is that many pairwise comparisons are required, and oftentimes the resulting data matrix is sparse, and average filling is required, which can compromise the reliability of results. They both tend to have individual tradeoffs, which will be further explored in a later section.

Hybrid based filtering, as proposed by Kumar and Fan combined both user based and item based filtering and also includes a more rigorous clustering step. It also involves the use of case based reasoning (CBR) which draws from past experiences and provides a better method of filling in null values to "densify" the matrix. This dense matrix is then fed into a genetic algorithm titled "self-organizing map" for identifying clusters of users with their given attributes and then top results are fed into an item-based filter.

How do each of these items work in practice?

Starting with Collaborative filtering this one works by analyzing the viewing patterns of multiple users. If two users have watched and enjoyed similar movies, they are considered similar.

The system then recommends movies watched by one user to the other based on shared preferences. For example, if one user frequently watches action movies and another user with similar tastes does the same, the system suggests movies from one to the other.

On the other hand, content-based filtering focuses on the characteristics of the movies themselves. If a user has watched a comedy movie, the system recommends other movies with similar attributes, such as genre, director, or cast.

The key difference between these approaches is that collaborative filtering relies on the preferences of other users with similar interests, while content-based filtering analyzes the movie's content to generate recommendations.

What are some notable differences between the user/iterm based content filtering and the hybrid filtering method being proposed?

For one, the hybrid approach has a more robust method of handling null values as evidenced by CBR vs average filling. This leads to better accuracy. Furthermore, because the hybrid method uses clustering, which greatly limits and narrows down the result set, hybrid filtering is far more scalable than user/item based content filtering, which rely heavily on pairwise comparisons. Another issue worth mentioning that affects both approaches is the problem of a cold start, where a user or item has just debuted, and there is little supporting information to go off of. With the hybrid approach, case based reasoning could potentially help fill in some of the gaps. User/Item based filtering can suffer from a problem of repetitive recommendations, as these are sourced solely from comparisons and not clusters.

In short, which approach is better?

In our opinion, user/item based content based filtering is ideal for systems with dense datasets, smaller user bases, and situations where simplicity and quick implementation is prioritized. This approach will work best when historical user-item interaction data is rich and consistent. If not, the pairwise comparisons will struggle. We recommend hybrid collaborative filtering for situations where the data is larger-scale, the dataset is more sparse, and there are greater concerns of scalability. Depending on the application, hybrid options may be preferred for modern, dynamic functions such as a streaming service like Netflix, where more real-time and accurate recommendations are expected.

Improvements for both methods

For UBCF + IBCF, we propose several improvements to enhance recommendation quality. One key area is deep learning integration—by using advanced neural networks like autoencoders or transformers, we can better capture the relationships between users and items, making recommendations more accurate.

Another improvement involves refining feedback profiles by incorporating implicit signals, such as how long users engage with content or how often they interact with certain items. This approach moves beyond simple ratings and provides a richer understanding of user preferences. Additionally, expanding the use of knowledge graphs can help establish deeper connections between users and items by integrating external data, leading to more personalized and explainable recommendations.

Lastly, addressing the cold start problem where new users or items lack enough data for accurate predictions can be tackled by incorporating extra information like demographics, browsing history, or contextual details such as time and location.

For the Hybrid approach, we suggest optimizing clustering techniques to create better user groups and improve recommendations for each segment. Advanced methods like spectral or deep clustering can help identify patterns in user behavior more effectively. We also recommend enhancing data densification, which involves filling in gaps in sparse data using methods like matrix factorization or graph-based techniques; this is especially useful when there’s limited interaction data available.

Additionally, knowledge graphs can be leveraged more effectively by using multi-relational embeddings or graph neural networks (GNNs). This allows the recommendation system to model complex relationships between users, items, and contextual factors, making recommendations not only more relevant but also more transparent and interpretable.

References

[1] N. Kumar, Z. Fan, “Hybrid User-Item Based Collaborative Filtering,” *Procedia Computer Science*, vol. 60, pp. 1453-1461, 2016. [Online]. Available:https://www.sciencedirect.com/science/article/pii/S1877050915023492.

[2] H. Wang, Z. Shen, S. Jiang, G. Sun, R. Zhang, “User-based Collaborative Filtering Algorithm Design and Implementation” *Journal of Physics,* 2020. [Online]/ Available: doi:10.1088/1742-6596/1757/1/012168

[3] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, *"Item-Based Collaborative Filtering Recommendation Algorithms,"* in Proceedings of the 10th International Conference on World Wide Web (WWW’01), Hong Kong, 2001, pp. 285–295.

[4] J. S. Breese, D. Heckerman, and C. Kadie, *"Empirical Analysis of Predictive Algorithms for Collaborative Filtering,"* in Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI’98), Madison, WI, USA, 1998, pp. 43-52.