Data Mining
Challenge Report

Nikolaos Efthymiou[1] and Luca Pantea[2]

[1]5066441, Kaggle ID: Nikos Efthymiou
[2]5040582, Kaggle ID: lucapantea

January 23, 2021

Kaggle Team Name: alligators

1 Introduction

The goal of the Netflix Challenge Project is to make accurate predictions for the ratings of users for movies. That is, to estimate whether someone will enjoy a movie based on how much they liked or disliked other movies. The data for this project contains 910,190 ratings (on a 1-5 scale) that were given by 6,040 users to 3,706 movies. Data for the users (gender, age and profession) and movies (title and year of release) is also provided.

We used a variety of different algorithms to make predictions. Some of the methods we used are the following: Collaborative Filtering (User-base and Item-based), Matrix Factorization (using Singular Value Decomposition and Stochastic Gradient Descent), global baseline estimates, demographics for users and content-based approaches for movies (item and user profiling respectively). In the following section, we will explain the nature of each algorithm and the motivation behind it.

2 Methodology

2.1 Collaborative Filtering (CF)

The most common form of CF is the neighborhood-based approach (kNN). Its intuitiveness and interpretability are the main reasons for its wide popularity. This approach explores similarities among target objects based on past ratings. We used two different variants of this algorithm (as shown below), where the predicted rating is computed using a weighted average over the ratings of the neighbouring users and movies accordingly ($k = 10$) [2, 3, 16, 10]:

- *User-Based Collaborative Filtering* is a technique used to predict the rating of a user for a particular movie based on the similarity among users according to past ratings for that movie (we used the *centred cosine similarity*). In particular, the predicted rating is computed as a weighted average over the ratings of the 10 nearest neighbours of the target user.
- *Item-Based Collaborative Filtering* exploits the similarity among movies (items) rather than users. Relevant research has shown that item similarity is more meaningful than user similarity as it can be inferred with greater confidence from the 'objective' attributes of items [10]. Conceptually, item similarity results to items being classified to specific genres or types more reasonably than user similarity, which infers subjective tastes, personality traits or demographic characteristics.

2.2 Matrix Factorization using Singular Value Decomposition

Latent factor models assume an underlying (hidden) low-dimensional representation of the correlations between movies and users, which can be inferred through techniques of matrix factorization (MF). More specifically, by using Singular Value Decomposition (SVD), we reduce the dimensionality of the original feature space to k factors ($k = 40$)[11]. These factors assume latent predictive variables like user and movie characteristics (user tastes, movie genres, demographics, etc.), that could affect the ratings.
As SVD fails to tackle missing values, we first normalized the utility matrix having the row values centered around 0 and then we added the user's average rating (that is, *reverting* the normalization) such that the resulting predictions are made on the original scale.

2.3 Global Baseline Estimation

After addressing *regional* effects through MF and extracting local patterns by means of CF, we still lack some global, top-level view of the data [15]. Thus, we improved our methods explained above using a global baseline estimate b_{xi}, corresponding to user x and movie i, which is computed as follows:

$$b_{xi} = \mu_x + \mu_i - \mu, \tag{1}$$

where μ is the average rating over all users and all movies and μ_x and μ_i are the average ratings of user x and movie i respectively. In this way, the prediction takes into account the deviation of the user and movie of interest from the overall average rating. So, a strict user tends to give low ratings, a blockbuster movie gets high ratings, etc.

2.4 Demographic and Content-based methods

We also applied a hybrid approach that builds upon CF to improve its predictability by combining information extracted from demographic data (Figure 1) and content (movie) data [17, 12, 1, 5]. Thus, we use the notion of an *enhanced similarity* which in case of the demographic-based method is defined in the following way:

$$enh\_sim_{xy} = sim_{xy} \cdot dem\_sim_{xy}, \tag{2}$$

where dem_sim_{xy} is the demographic similarity between the active user x and a neighbouring user y and sim_{xy} is the similarity between the two users based on past ratings (as used in the base CF approach). Specifically, we compare the users according to their age using the following formula:

$$dem\_sim_{xy} = 1 - \frac{|age_x - age_y|}{max\_age - min\_age} \tag{3}$$

The predicted rating for user u and movie i is then computed as follows:

$$\hat{r}_{ui} = \frac{\sum_{j=1}^{k} r_{ji} \cdot enh\_sim_{uj}}{\sum_{j=1}^{k} |enh\_sim_{uj}|}, \tag{4}$$

where k is the cardinality of the neighbourhood of the target user.

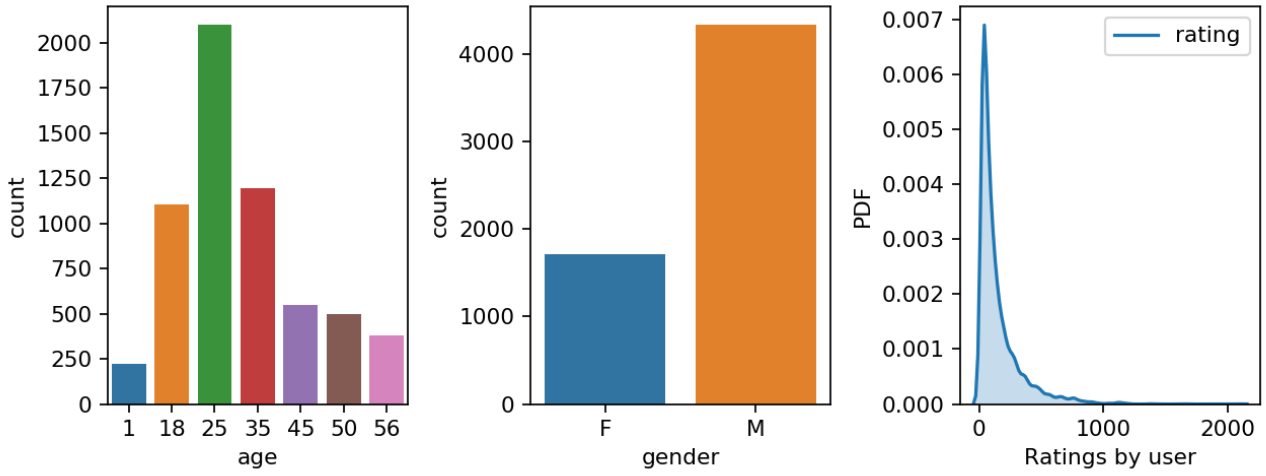

Figure 1: Statistics obtained from the given dataset

The above analyzed enhanced similarity method has been applied for the content-based approach as well. In this case, we used the movie's year of release to compute $cont\_sim_{xy}$ as follows:

$$cont\_sim_{xy} = 1 - \frac{|year_x - year_y|}{max\_year - min\_year} \tag{5}$$

2.4.1 IMDb Dataset Integration

Since our previously discussed algorithms were constructing the recommendation model based on the 2 main features of the original movie dataset, namely the title and the year of a given movie, we set out to find new ways of adding

more reference points in our data. After considerable research, we turned towards the extensive IMDb dataset [9] which provided us with the opportunity to take factors such as the genre or the metascore of a given movie into account. To find a reliable one-to-one mapping between the two datasets, we did a *left join*, which grew our feature space significantly. Due to time constrains we decided to investigate only the genres of the movies, representing each movie k as a vector $v \in \{0,1\}^q$ (q different genres), where $v_i = 1$ if and only if movie k belongs to genre i. We modified the way we calculate $cont\_sim_{xy}$ in the following way:

$$cont\_sim_{xy} = \frac{q - d(x,y)}{q} \tag{6}$$

where $d(x,y)$ is the *hamming distance* between the vector representation of movies x and y.

2.5 Matrix Factorization using Stochastic Gradient Descent (SGD)

SGD-based factorization has been a successful method for recommendation algorithms and it was used by the winners of the Netflix challenge [6, 4]. The rating estimation equation for the SGD method can be written as:

$$\hat{r}_{ui} = {q_i}^T \cdot p_u, \tag{7}$$

where $q_i, p_u \in \mathbb{R}^k$ describe movie i and user u respectively. k is the dimensionality of the latent space and we want to find matrices $\underset{(3706\times k)}{Q}$ and $\underset{(k\times 6040)}{P}$ that map movies and users to that latent space. As a good initial point, these matrices are initialized using SVD, pretending missing values are 0. Then, given the prediction error $e_{ui} = 2 \cdot (r_{ui} - q_i \cdot p_u)$, the values of q_i, p_u are updated in an iterative fashion (we used 15 epochs) following the equations:

$$\begin{cases} q_i := q_i + m_1 \cdot (e_{ui} \cdot p_u - l_2 \cdot q_i) & \text{(8a)} \\ p_u := p_u + m_2 \cdot (e_{ui} \cdot q_i - l_1 \cdot p_u) & \text{(8b)} \end{cases}$$

where l_1, l_2 are the regularization constants with value 0.07, m_1, m_2 are the learning rates with value 0.005 and $k = 40$. [11, 7]
In the methodological framework of our hybrid approach (see above), we used the content-based enhanced CF (item-item) in combination with SGD to compute the final predicted rating $\hat{r}_{ui}$ in the following way:

$$\hat{r}_{ui} = \alpha \cdot \frac{\sum_{j=1}^{k} r_{uj} \cdot enh\_sim_{ij}}{\sum_{j=1}^{k} |enh\_sim_{ij}|} + (1-\alpha) \cdot {q_i}^T \cdot p_u \tag{9}$$

3 Results

In order to compare the performance of the algorithms applied, we computed the Root Mean Sqaured Error (RMSE), which is broadly used as an evaluation metric in recommender systems. For each algorithm, we performed *hyper-parameter tuning* to choose the values for certain parameters that minimize the RMSE score. In Figure 2 we present the best (lowest) attained scores for each method. According to these results, the most effective method was the **hybrid SGD-enhanced-content-CF**. After experimenting with different weights for α (see equation 9), the best score was attained for $\alpha = 0.35$.

4 Discussion and Conclusions

Comparing the RMSE scores of the methods, we can make several observations regarding the inner workings of our chosen algorithms. The item-based CF approaches outperformed the user-based in all tested cases. This was the result we were expecting, as explained both in 2.1, and touched upon in the lectures of the course. Yet another result that stands out is that the implementation of the global baseline estimations improves the performance of the algorithms, especially in the case of user-based CF, as shown in figure 2. Moreover, an additional conclusion to be drawn is that the enhanced versions of CF outperform the base versions. This was expected, as the former exploits more data representative of the target objects than the latter.

One of the most interesting outcomes is the great improvement the hybrid approaches brought to the overall performance of the algorithms. We turned to hybrid approaches to both alleviate the shortcomings of our methodologies, and to explore how different algorithms behave when combined together (*Ensemble Learning*). Through this framework, we managed to vastly improve our understanding of the dataset, which, on the one hand, allowed us

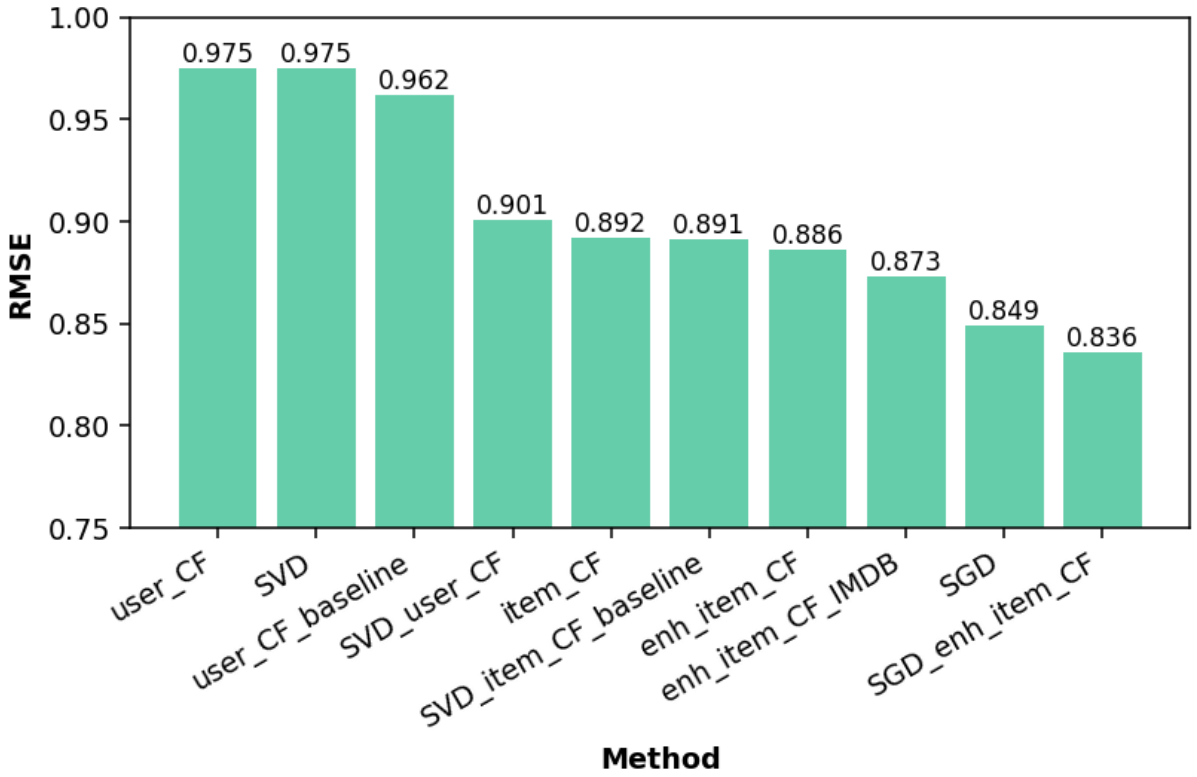

Figure 2: The performance of our algorithms, measured with $RMSE$

to make more impactful decisions regarding the content-based approaches (Subsection 2.4), and on the other hand, made us even more intrigued to further explore with other methodologies and new datasets.

Finally, by integrating the extensive IMDb dataset, we managed to obtain key information on the movies given. Though our implementation only took account of genre of each movie, which gave us some improvements to our content-based algorithms (as shown in Figure 2), we acknowledge that other features can be exploited in future projects to yield more accurate predictions (e.g. critique's ratings, date of ratings, etc.).

5 Future Work

Even though the RMSE scores were satisfactory with the methods presented in our report, there are still numerous alternative approaches to be considered. Different matrix factorization (MF) techniques can be explored, such as *Alternating Least Squares* (ALS), *Biased Regularized Incremental Simultaneous MF* (BRISMF), Non-negative MF and Deep MF models [14, 13, 19, 18]. The *temporal dynamics* (time-aware factor models) of user and item biases are something that can further improve the performance of the algorithms [8].

References

[1] Gharbi Alshammari et al. "Improved Movie Recommendations Based on a Hybrid Feature Combination Method". In: *Vietnam Journal of Computer Science* 06.03 (2019), pp. 363–376. DOI: 10.1142/S2196888819500192.

[2] Mojdeh Bahadorpour, Behzad Soleimani Neysiani, and Mohammad Nadimi Shahraki. "Determining Optimal Number of Neighbors in Item-based kNN Collaborative Filtering Algorithm for Learning Preferences of New Users". In: *Journal of Telecommunication, Electronic and Computer Engineering* 9.3 (2017), pp. 163–167.

[3] R. Bell and Y. Koren. "Improved Neighborhood-based Collaborative Filtering". In: 2007. URL: https://www.cs.uic.edu/~liub/KDD-cup-2007/proceedings/Neighbor-Koren.pdf.

[4] J. Bennett and S. Lanning. "The Netflix Prize. Proceedings of KDD Cup and Workshop 2007, Aug. 12, 2007". In: 2007. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.117.8094&rep=rep1&type=pdf.

[5] Robin Burke. "Hybrid Recommender Systems: Survey and Experiments". In: *User Modeling and User-Adapted Interaction* 12.04 (2002), pp. 331–370. DOI: 10.1023/A:1021240730564.

[6] Gideon Dror et al. "The Yahoo! Music Dataset and KDD-Cup'11". In: *Proceedings of KDD Cup 2011*. Ed. by Gideon Dror, Yehuda Koren, and Markus Weimer. Vol. 18. Proceedings of Machine Learning Research. PMLR, 21 Aug 2012, pp. 3–18. URL: http://proceedings.mlr.press/v18/dror12a.html.

[7] Y. Koren, R. Bell, and C. Volinsky. "Matrix Factorization Techniques for Recommender Systems". In: *Computer* 42.8 (2009), pp. 30–37. DOI: 10.1109/MC.2009.263.

[8] Yehuda Koren. "Collaborative Filtering with Temporal Dynamics". In: KDD '09. Paris, France: Association for Computing Machinery, 2009, pp. 447–456. DOI: 10.1145/1557019.1557072.

[9] Stefano Leone. *IMDb movies extensive dataset*. URL: https://www.kaggle.com/stefanoleone992/imdb-extensive-dataset/notebooks. (accessed: 19.01.2021).

[10] Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. *Mining of Massive Datasets*. 2nd ed. Cambridge University Press, 2014, pp. 321–325. DOI: 10.1017/CBO9781139924801.

[11] Aanchal Mongia et al. "Deep latent factor model for collaborative filtering". In: *Signal Processing* 169 (2020), p. 107366. DOI: https://doi.org/10.1016/j.sigpro.2019.107366.

[12] Michael J. Pazzani. "A Framework for Collaborative, Content-Based and Demographic Filtering". In: *Artificial Intelligence Review* 13.5 (Dec. 1999), pp. 393–408. DOI: 10.1023/A:1006544522159.

[13] G. Takács et al. "Investigation of Various Matrix Factorization Methods for Large Recommender Systems". In: *2008 IEEE International Conference on Data Mining Workshops*. 2008, pp. 553–562. DOI: 10.1109/ICDMW.2008.86.

[14] Gábor Takács and Domonkos Tikk. "Alternating Least Squares for Personalized Ranking". In: RecSys '12. New York, NY, USA: Association for Computing Machinery, 2012, pp. 83–90. ISBN: 9781450312707. DOI: 10.1145/2365952.2365972.

[15] Zhenhua Tan et al. "Personalized Standard Deviations Improve the Baseline Estimation of Collaborative Filtering Recommendation". In: *Applied Sciences* 10.14 (2020). DOI: 10.3390/app10144756.

[16] M.G. Vozalis and K.G. Margaritis. "Using SVD and demographic data for the enhancement of generalized Collaborative Filtering". In: *Information Sciences* 177.15 (2007), pp. 3017–3037. DOI: https://doi.org/10.1016/j.ins.2007.02.036.

[17] Manolis Vozalis and Konstantinos G. Margaritis. "Collaborative Filtering enhanced by Demographic Correlation. AI Symp. Prof. Pract. AI, 18th World Comput. Congr." In: (Jan. 2004), pp. 1–10. URL: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.8507&rep=rep1&type=pdf.

[18] Hong-Jian Xue et al. "Deep Matrix Factorization Models for Recommender Systems". In: *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 2017, pp. 3203–3209. DOI: 10.24963/ijcai.2017/447.

[19] Sheng Zhang et al. "Learning from Incomplete Ratings Using Non-negative Matrix Factorization". In: *Proceedings of the 2006 SIAM International Conference on Data Mining*, pp. 549–553. DOI: 10.1137/1.9781611972764.58.