NEWS
OF THENATIONAL ACADEMY OF SCIENCES OF THE REPUBLIC OF KAZAKHSTAN
PHYSICO-MATHEMATICAL SERIES
ISSN 1991-346X
Volume 1, Number 335 (2021), 26 – 31 **https://doi.org/10.32014/2021.2518-1726.4**

UDC 004.89
IRSTI 28.23.01

A.Y. Zhubatkhan, Z.A. Buribayev,
S.S. Aubakirov, M.D. Dilmagambetova, S.A. Ryskulbek

Al-Farabi Kazakh National University, Almaty, Kazakhstan.
E-mail: d.m.d97@mail.ru

COMPARISON MODELS OF MACHINE LEARNING FOR MOVIE RECOMMENDATION SYSTEMS

Abstract. The trend of the Internet makes the presentation of the right content for the right user inevitable. To this end, recommendation systems are used in areas such as music, books, movies, travel planning, e-commerce, education, and more. One of the most popular recommendation systems in the world is Netflix, which generated record profits during quarantine in the first quartile of 2020. The systematic approach of recommendations is based on the history of user selections, likes and reviews, each of which is interpreted to predict future user selections. This article provides a meaningful analysis of various recommendation systems, such as content-based, collaborative filtering and popularity. We reviewed 7 articles published from 2005 to 2019 to discuss issues related to existing models. The purpose of this article is to compare machine learning algorithms in the Surprise library for a recommendation system. Recommendation system has been implemented and quality has been evaluated using the MAE and RMSE metrics.

Keywords: recommendation system, analysis of machine learning approaches, Surprise library, collaborative filtering.

1. Introduction

Recommender systems are algorithms designed to offer users the appropriate elements. Recommender systems are a class of information filtering systems whose main purpose is to provide personalized recommendations, content and services to users. Recommender systems typically help users find products such as films, books, articles, news, and others that match their personal preferences and needs [1].

Personalized recommendation blocks are the most obvious example of user personalization. The web service ranks the objects in order of relevance for a particular user based on user history. For example, on Netflix service, rough estimates posted continuous content [2]. The user is not able to watch the whole movie, so users search for a movie that they like by using stories at the same time. The task of this system is to build a personal stream that is interesting to the user based on the website.

Recommender systems are not necessarily intended to recommend certain objects to users. To increase the effectiveness of promotions, online stores resort to the help of recommendation systems in order to identify the most interested users in one of the products. Recommendation system predicts the degree of interest of each user to a particular product based on purchases and their responses to promotional letters [3].

Referral systems are really important in some industries, as they can generate huge revenues when they are effective. As evidence of the importance of recommender systems, we can mention that a few years ago Netflix organized a contest with a prize of $ 1 million, where the goal was to create a recommendation system that works better than its own algorithm [4].

Recommender systems work with two types of information:

1. Characteristic information. This is information about elements and users.
2. User interaction. This information is such as ratings, number of purchases, likes, etc.

Based on this, we can distinguish three algorithms used in recommender systems:

1. Content systems that use specific information.
2. Collaborative filtering systems based on user interaction with the element.
3. Hybrid systems that combine both types of information in order to avoid problems that arise when working with only one type.

Collaborative methods of recommendation systems are methods that are based on past interactions recorded between users and subjects to develop new recommendations. These interactions are stored in the so-called "user-element interaction matrix" [3]. Then, the basic idea that governs collaborative methods is that these past user-element interactions are enough to detect similar users or similar elements and predict based on these assumed approximations. However, since only past interactions are taken into account for making recommendations, collaborative filtering suffers from a “cold start problem”: it is not possible to recommend something to new users or recommend a new item to any users, and many users or elements have few interactions too. This drawback can be eliminated in different ways: recommend random items to new users or new items to random users, recommend popular items to new users or new items to most active users, recommend a set of different items for new users or a new item to recruit different users.

Unlike collaborative work methods that rely on the interaction of user elements, content-based approaches use additional information about users or elements. If we look at an example of a movie recommendation system, this additional information could be, for example, age, gender, work or any other personal information for users, as well as the category, main characters, duration or other characteristics for movies [5].

Then the idea of content-based methods is to try to build a model based on the available “functions” that explain the observed user interactions with the element.

Content-based methods suffer much less from the “cold start” problem than collaborative approaches [6]. Only new users or elements with previously unseen features will logically suffer from this shortcoming, but as soon as the system becomes old enough, it will have little chance of not happening at all.

2. Methods

To develop recommendation system algorithms, used the Surprise library, which was built by Nicolas Hug. Surprise is a library in Python scikit for recommender system, which is able to build an algorithm, that is nothing but a class derived from “AlgoBase” that has an “estimate” method. This is the method that is called by the predict() method. It takes in an inner user id, an inner item id, and returns the estimated rating r_{ui} . But the dumbest algorithm returns a set rating value.

In this research used MovieLens datasets by the GroupLens Research Project at the University of Minnesota. Dataset consists of 100,000 ratings (1-5) from 943 users on 1682 movies. Each user has rated at least 20 movies [7].

To fit prediction algorithms, it requires a similarity measure, which builds a similarity matrix and returns value depending on the similarity of films and users.

To make a cleverer algorithm that predicts the average of all the ratings of the train set. As this is a constant value that does not depend on current user or item, we would rather compute it once and for all. This can be done by defining the fit method. This way, we can fit our algorithms for training sets.

To prediction used the prediction algorithms package of Surprise library, which includes the prediction algorithms available for recommendation. We used nine type of prediction algorithm:

1. NormalPredictor is an algorithm which predicts a random rating based on the distribution of the training set, which is assumed to be normal.
2. BaselineOnly is an algorithm which predicts the baseline estimate for a given user and item.
3. KNNBasic is a basic collaborative filtering algorithm.
4. KNNWithMeans is basic collaborative filtering algorithm, taking into account the mean ratings of each user.
5. KNNBaseline is a basic collaborative filtering algorithm taking into account a baseline rating.
6. SVD is equivalent to Probabilistic Matrix Factorization.
7. NMF is a collaborative filtering algorithm based on Non-Negative Matrix Factorization. It is very similar with SVD.

8. SlopeOne is a straightforward implementation of the SlopeOne algorithm. Coclustering is a collaborative filtering algorithm based on co-clustering.

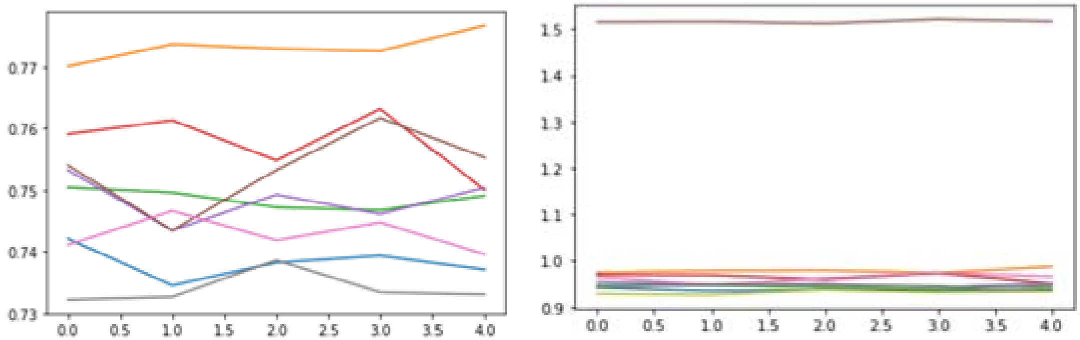
At the end, calculated evaluation metrics root mean square error (RMSE) and mean absolute error (MAE) on a 5-fold cross-validation procedure by formula (1) and (2). The folds are the same for all the algorithms [8].

$$MAE = \frac{1}{N}\sum_{i=1}^{N} | y_i - \hat{y} | \tag{1}$$

$$RMSE = \sqrt{MAE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} | y_i - \hat{y} |}$$

3. Results

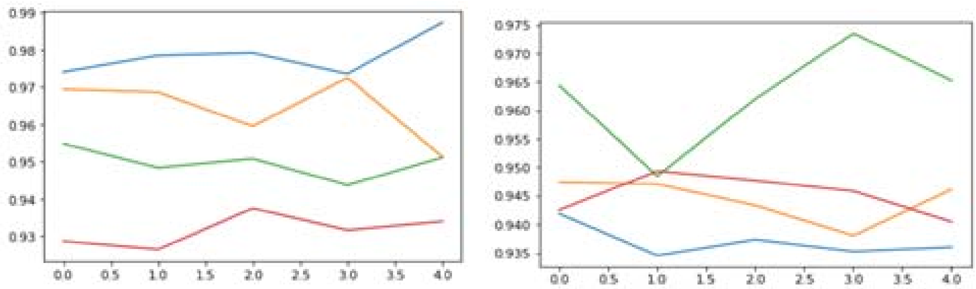
Nine types of machine learning models showed different results for evaluation of quality: MAE and RMSE [9]. Training is conducted for datasets MovieLens 100k to compare the performance of machine learning algorithms: SVD, KNN, KNNwithMeans, KNNBasic, BaselineOnly, Coclustering, SlopeOne, NMF and Normal Predictor. First, the overall results of the MAE and RMSE for all approaches in Figure 1.



(a) (b)

Figure 1 - Results: a - results of MAE for each approach, b - results of RMSE for each approach

Here, brown line – Normal Predictor, orange line is KNN, red line is NMF (Not negative matrix factorization), violet line is KNN with Means and grey blue line for KNN Basic, dark green line is CoClustering, blue line is BaselineOnly, green line is SVD (Support vector machine) and salad green line is for SlopeOne.

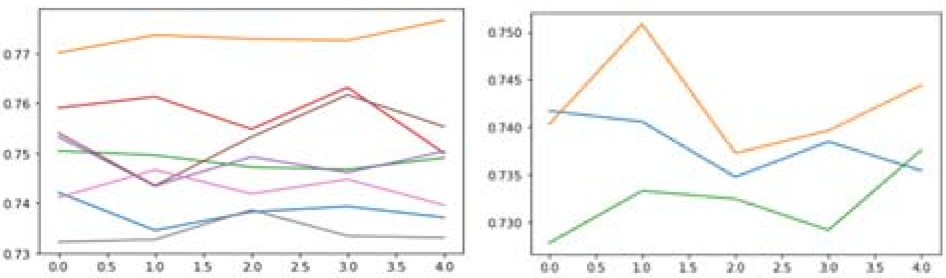


(a) (b)

Figure 2 - RMSE results. a - KNN, KNN with Mean, KNN Basic and NMF and b - SVD, BaselineOnly, Coclustering, SlopeOne

Figure 2.a shows result of RMSE for some algorithms, which are better than Normal Predictor. Here, blue line is KNN, orange line is NMF, green line is KNN with Means and red line is KNN Basic. In Figure 2.b shows top results of RMSE for four algorithms: SVD, BaselineOnly, Coclustering, SlopeOne. Here, green line is Coclustering, red line is BaselineOnly, orange line is SVD and blue line is for SlopeOne.

Figure 3 shows results of MAE for all approaches without Normal Predictor:



(a) (b)

Figure 3 - MAE results: a - all approaches, b - good results

In 3.a, orange line is KNN, red line is NMF, brown line KNN Basic, BaselineOnly by violet line, green line is Coclustering, pink line is SlopeOne, blue line is SVD, grey line KNN with Mean. Mean absolute error returns good results for two approaches: KNN with Means and SlopeOne. Figure 3.b in scaled view for approaches with good result:

Here, the orange line is SlopeOne, the blue line is SVD, the line is for KNN with Means. As we can see that the best results of MAE are KNN with Means and SVD predictions algorithms.

4.Conclusion. Recommendation systems make convenient Internet by predicting the right content for the right user. But also gives the other problem such as: what kind of approach to use, indeed this method is inevitable for a given dataset? This paper discussed the nine traditional approaches and highlighted their advantages and disadvantages by evaluation mean absolute error and root-mean-square error. By measuring the quality closed SVD and KNNwM algorithms predicted with minimal mean absolute error (MAE), but by root-mean-square error (RMSE) the best predictors are SVD and SlopeOne. In table 1 shown prediction algorithms with values of MAE for spited into 5-fold cross validation.

Table 1 - MAE results of the best approaches for given results

Algorithms	Fold 1	Fold 2	Fold 3	Fold 4	Ford 5	Mean	Std
SVD	0.7417	0.7405	0.7347	0.7384	0.7354	0.7381	0.0037
KNN with Means	0.7501	0.7430	0.7528	0.7511	0.7485	0.7491	0.0034

As we can see that the best results of RMSE are BaselineOnly and SlopeOne predictions algorithms. In Table 2 shown prediction algorithms with values of RMSE for spited into 5-fold cross validation:

Table 2 - RMSE results of the best approaches for given results

Algorithms	Fold 1	Fold 2	Fold 3	Fold 4	Ford 5	Mean	Std
SVD	0.9418	0.9345	0.9373	0.9353	0.936	0.937	0.0085
SlopeOne	0.9425	0.9493	0.9477	0.9458	0.9405	0.9452	0.0033

А.Ы. Жұбатхан, Ж.А. Бурибаев,
С.С. Аубакиров, М. Д. Дильмагамбетова, С.А. Рысқұлбек

Әл-Фараби атындағы ҚазҰУ, Алматы, Қазақстан

ФИЛЬМ ҰСЫНУ ЖҮЙЕЛЕРІНЕ АРНАЛҒАН МАШИНАЛЫҚ ОҚЫТУ МОДЕЛЬДЕРІН САЛЫСТЫРУ

Аннотация. Қазіргі интернет тенденциясы қолданушыға нақты әрі қажетті контентті ұсынуды сөзсіз орындайды. Осы мақсатта ұсыныс жүйелері музыка, кітап, фильм, саяхат жоспарлау, электронды сауда, білім беру және тағы басқа салаларда қолданылады. Әлемдегі ең танымал ұсыныс жүйелерінің бірі – Netflix. Бұл карантин кезінде 2020 жылдың бірінші маусымында пайда түсіру жағынан рекордқа жетті.

Ұсыныстарға жүйелі көзқарас пайдаланушылардың таңдауының, ұнатуының және шолуларының тарихына негізделген, олардың әрқайсысы болашақ пайдаланушылардың сайлауын болжау ретінде түсіндіріледі.

Мақала мазмұнға, бірлескен сүзгілеу және танымалдық секілді түрлі ұсыныс жүйелерінің талдамасына негізделген. Қолданыстағы модельдерге қатысты мәселелерді талқылау үшін 2005 жылдан 2019 жылға дейін жарияланған 7 мақаланы қарастырдық. Мақаланың мақсаты – ұсыныстар жүйесі үшін Surprise кітапханасындағы машиналық оқыту алгоритмдерін салыстыру. Ұсыныстар жүйесі бағдарламаланды әрі MAE және RMSE сапа көрсеткіштерін қолдану арқылы бағаланды.

Түйін сөздер: ұсыныстар жүйесі, машиналық оқыту тәсілдерін талдау, Surprise кітапханасы, бірлескен сүзгілеу әдісі.

А.Ы. Жұбатхан, Ж.А. Бурибаев,
С.С. Аубакиров, М. Д. Дильмагамбетова, С.А. Рысқұлбек

Казахский национальный университет им. аль-Фараби, Алматы

СРАВНЕНИЕ МОДЕЛИ МАШИННОГО ОБУЧЕНИЯ ДЛЯ СИСТЕМ РЕКОМЕНДАЦИЙ ФИЛЬМОВ

Аннотация. Интернет-тенденция делает неизбежной презентацию нужного контента для нужного пользователя. С этой целью рекомендательные системы используются в таких областях, как музыка, книги, фильмы, планирование путешествий, электронная коммерция, образование и т.д. Одна из самых популярных систем рекомендаций в мире – Netflix, которая принесла рекордную прибыль во время карантина в первом квартале 2020 года.

Систематический подход к рекомендациям основан на истории пользовательских выборов, лайков и обзоров, каждая из которых интерпретируется как предсказатель будущих выборов пользователей.

В этой статье представлен содержательный анализ различных систем рекомендаций, таких как контентная, совместная фильтрация и популярность. Мы просмотрели 7 статей, опубликованных с 2005 по 2019 год, чтобы обсудить вопросы, связанные с существующими моделями. Цель этой статьи - сравнить алгоритмы машинного обучения в библиотеке Surprise для рекомендательной системы. Внедрена система рекомендаций, и качество оценено с использованием показателей функций MAE и RMSE.

Ключевые слова: рекомендательные системы, анализ подходов к машинному обучению, библиотека сюрпризов, коллаборативная фильтрация.

Information about authors:

Zhubatkhan A.Y., Master's degree student, Al-Farabi Kazakh National University, Almaty, Kazakhstan; aishazhy19@gmail.com, https://orcid.org/0000-0003-0292-2212;

Buribayev Z.A., Master of Science, Senior Lecturer, Deputy Head of the Department for Scientific and Innovative Work and International Relations, Al-Farabi Kazakh National University, Almaty, Kazakhstan; https://orcid.org/0000-0002-3486-227X;

Aubakirov S.S., PhD, Senior Lecturer, Al-Farabi Kazakh National University, Almaty, Kazakhstan; c0rp.aubakirov@gmail.com, https://orcid.org/0000-0002-8416-527X;

Dilmagambetova M.D., Master's degree student, Al-Farabi Kazakh National University, Almaty, Kazakhstan; d.m.d97@mail.ru, https://orcid.org/0000-0002-8456-5417;

Ryskulbek S.A., Master's degree student, Al-Farabi Kazakh National University, Almaty, Kazakhstan; sannyrys9@gmail.com, https://orcid.org/0000-0002-8711-4398

REFERENCES

[1] Beel J.: Towards effective research-paper recommender systems and user modeling based on mind maps. PhD Thesis. Otto-vonGuericke Universität Magdeburg (2015).

[2] Paraschiv and Ionut Cristian. A Paper Recommendation System with ReaderBench: The Graphical Visualization of Semantically Related Papers and Concepts, In State-of-the-Art and Future Directions of Smart Learning, pp. 445–451, (2016).

[3] Bhagirathi Nayak, Rajesh K. Ojha, P. S. Subbarao, VijayaBatth. Machine Learning Finance: Application of Machine Learning in Collaborative Filtering Recommendation System for Financial Recommendations // International Journal of Recent Technology and Engineering (IJRTE) // ISSN: 2277–3878, p. 8, Issue-1, May 2019.

[4] Ibukun Afolabi. A Model for Business Success Prediction using Machine Learning Algorithms // Journal of Physics Conference Series, 1299:012050, August.

[5] Claire Longo. Joint Neural Collaborative Filtering for Recommender Systems // ACM Transactions on Information Systems // Publication date: July 2019.

[6] Ibukun Afolabi, T. Cordelia Ifunaya, Funmilayo G. Ojo, Chinonye Moses. A Model for Business Success Prediction using Machine learning algorithms // 3rd International Conference on Science and Sustainable Development // 2019. URL: https://www.researchgate.net/publication/336330683_A_Model_for_Business_Success_Prediction_using_Machine_Learning_Algorithms

[7] Mitchel T. Machine Learning. McGraw-Hill Education (ISE Editions). 1997.

[8] M. Pazzani and D. Billsus. Learning and Revising User Profiles: The identification of interesting web sites. Machine Learning, 27:313–331, 1997.

[9] The web site of the research lab in the Department of Computer Science and Engineering at the University of Minnesota. URL:https://grouplens.org/datasets/movielens (accessed: 10.11.2019 y.).

[10] Zehra C.¸ ATALTEPE, Mahiye ULUYAGMUR, Esengul TAYFUR. Feature selection for movie recommendation // Turkish Journal of Electrical Engineering & Computer Sciences // p. 833–848, March 2016.

[11] Paraschiv and Ionut Cristian, A Paper Recommendation System with ReaderBench: The Graphical Visualization of Semantically Related Papers and Concepts, In State-of-the-Art and Future Directions of Smart Learning, pp. 445–451, (2016).

[12] Beel J.: Towards effective research-paper recommender systems and user modeling based on mind maps. PhD Thesis. Otto-vonGuericke Universität Magdeburg (2015).

[13] Seroussi Y., Zukerman I., Bohnert F.: Collaborative inference of sentiments from texts. In: De Bra, P., Kobsa, A., Chin, D. (eds.) User Modeling, Adaptation, and Personalization, pp. 195–206. Springer, Berlin (2010).

[14] Brooks T.A.: Private acts and public objects: an investigation of citer motivations. J. Am. Soc. Inf. Sci. 36(4), 223–229 (1985).

[15] Carmagnola F., Cena F., Gena C.: User model interoperability: a survey. User Model. User-Adapt. Interact. 21(3), 285–331 (2011).

[16] Burke R., Ramezani M.: Matching recommendation technologies and domains. In: Ricci F., Rokach L., Shapira B., Kantor P.B. (eds.) Recommender Systems Handbook, pp. 367–386. Springer (2011).

[17] Ozono T., Shintani T.: Paper classification for recommendation on research support system papits. IJCSNS Int. J. Comput. Sci.Netw. Secur. 6, 17–23 (2006).

[18] Gipp B., Beel J., Hentschel C.: Scienstein: a research paper recommender system. In: Proceedings of the international conference on Emerging trends in computing (ICETiC’09), pp. 309–315 (2009).

[19] Beel J., Langer S., Nürnberger A., Genzmehr M.: The Impact of Demographics (Age and Gender) and Other User Characteristics on Evaluating Recommender Systems. In: Proceedings of the 17th International Conference on Theory and Practice of Digital Libraries (TPDL 2013), pp. 400–404 (2013).

[20] Geyer-Schulz A., Hahsler M.: Comparing two recommender algorithms with the help of recommendations by peers. In: Proceedings of the WEBKDD 2002—Mining Web Data for Discovering Usage Patterns and Profiles, pp. 137–158 (2003).

[21] Kuberek M., Mönnich M.: Einsatz von Recommender systemen in Bibliotheken Recommender systems in libraries. Presentation (2012).

[22] Küçüktunç O., Saule E., Kaya K., Çatalyürek Ü.V.: Diversifying citation recommendations. arXiv preprint. arXiv:1209.5809. pp. 1–19 (2012).