

# 커뮤니티 탐지 및 병렬 프로그래밍을 이용한 영화 추천 시스템

일홈준<sup>1</sup>, 양예선<sup>3</sup>, 썩소니<sup>3</sup>, 잇소포호트<sup>1</sup>, 김대영<sup>2</sup>, 박두순<sup>2\*</sup>

<sup>1</sup>순천향대학교 소프트웨어융합학과 석사과정

<sup>2</sup>순천향대학교 컴퓨터소프트웨어학과 교수

<sup>3</sup>순천향대학교 소프트웨어융합학과 박사과정

[ilkhomjon.sadriddinov@gmail.com](mailto:ilkhomjon.sadriddinov@gmail.com)(일홈준), [dyoung.kim@sch.ac.kr](mailto:dyoung.kim@sch.ac.kr)(김대영), [parkds@sch.ac.kr](mailto:parkds@sch.ac.kr)(박두순)

## Movie Recommendation System using Community Detection and Parallel Programming

Sadriddinov Ilkhomjon<sup>1</sup>, Yixuan Yang<sup>1</sup>, Sony Peng<sup>1</sup>, Sophort Siet<sup>1</sup>, Dae-Young Kim<sup>2</sup>,  
and Doo-Soon Park<sup>2</sup>

<sup>1</sup>Dept. of Software Convergence, Soonchunhyang University

<sup>2</sup>Dept. of Computer Software Engineering, Soonchunhyang University

### Abstract

In the era of Big Data, humanity is facing a huge overflow of information. To overcome such an obstacle, many new cutting-edge technologies are being introduced. The movie recommendation system is also one such technology. To date, many theoretical and practical kinds of research have been conducted. Our research also focuses on the movie recommendation system by implementing methods from Social Network Analysis(SNA) and Parallel Programming. We applied the Girvan-Newman algorithm to detect communities of users, and a *future* package to perform the parallelization. This approach not only tries to improve the accuracy of the system but also accelerates the execution time. To do our experiment, we used the MovieLens Dataset.

### 1. Introduction

Research on Movie Recommendations became so hot topic when Netflix launched the competition in 2009. So far, extensive research papers were published and are being read by many other researchers. Several criteria identify the system's acceptance. For example, how fast does the system provide the result and how correct is that result? Additionally, how much space(or memory) does the system need to accomplish the given task? By considering these factors, in our paper, we tried to consider the accuracy and execution time of our algorithm.

### 2. Related Work

Recommendation System(RS) uses information filtering to provide products (information, movie, book, etc.) to a user based on his/her preference and desires. Such systems learn user's interests by different means. Based on this means, RSs can be divided into several kinds. Such as

Collaborative Filtering, Content-Based Filtering, and Hybrid Filtering. If RS learns the behavior of actions among users then this method is Collaborative Filtering[1]. This method is

\*Corresponding author: Doo-Soon Park, Dae-Young Kim

Acknowledgments: This research was supported by the National Research Foundation of Korea (No. NRF-2022R1A2C1005921) and BK21 FOUR (Fostering Outstanding Universities for Research) (No.5199990914048)

based on dynamic information since the behavior might be changed after a user performs a new action. On the other hand, Content-Based Filtering focuses on a more static set of information. More precisely, it uses descriptive information about the user or a product[2]. The Hybrid Filtering approach is simply by its name a combined version of the above two techniques[3].

Social Network Analysis(SNA) is a domain where the relationship among people in groups(or communities) is considered. Many problems can be solved using techniques from SNA; such as finding the most important person(or an actor) inside the community, finding a group of people who share some commonalities, and others. In this research, we mainly focused on the community detection algorithms, namely, the Girvan-Newman algorithm[4]. This method measures the edge betweenness values for all edges in a network. After that, removes the edge with the highest edge betweenness value. The aforementioned steps are repeated iteratively until the desired community is detected. The concept known as Modularity can determine how well the community is divided[5].

Nowadays, modern gadgets(laptops, desktops, servers) are using multicore processors. The expensive(in time) algorithms are leveraged by the presence of Parallel Programming concepts. So in our paper, we also implemented the concepts of parallelism and concurrency[6]. The core idea about this is that parallel computing can increase system performance by executing the sub-tasks on multiple processors. There are many APIs (Application Program Interfaces) such as MPI, Pthreads, and OpenMP.

### 3. Design for Movie Recommendation System

The environment for the experiment is shown on the following Table 1.

Figure 1. illustrates the architecture of our experiment.

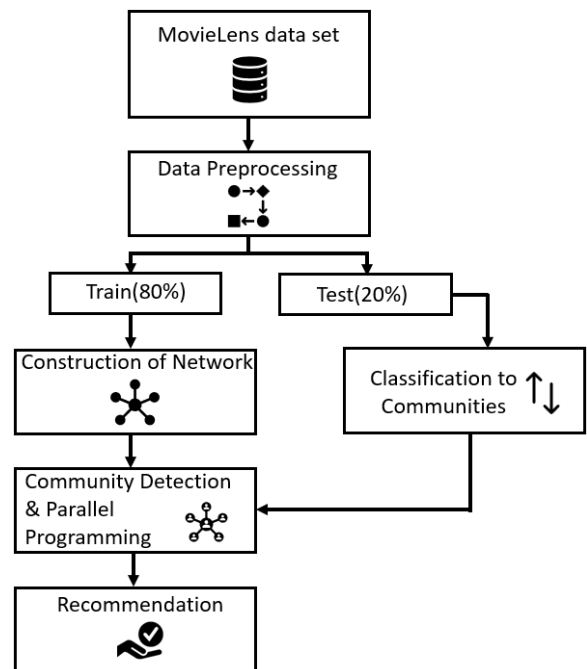
As the first step, we prepared our dataset for the next successive stages. We selected only three attributes: user age, gender, and occupation. All attributes were discretized to perform similarity measurements among users. Age values are categorized into 5 different

intervals[7].

<Table 1> Experimental Environment

Division	Detailed contents
CPU	12th Gen Intel(R) Core(TM) i7-12700K, 3.60 GHz
RAM	32GB
SSD	512GB
OS	Windows 11 Home 64bit
DEV.TOOL	R version 4.2.2 (2022-10-31) RStudio Version 2022.12.0

We also minimized the number of occupations by grouping similar jobs[8].



(Figure 1) The architecture of the Movie Recommendation System.

We compared three values for each user and built a binary adjacency matrix. If all values were identical for both users we assigned 1 to that particular entity otherwise, 0.

Once we finished the previous step, we applied a Girvan-Newman method to detect communities from the undirected graph. At the same time, we applied parallelism as well. By implementing a *future* package we improved the execution time for our system.

The algorithm detected 24 communities in total. And we removed the communities that contained fewer than 6 members.

#### 4. Experiment and Result Analysis

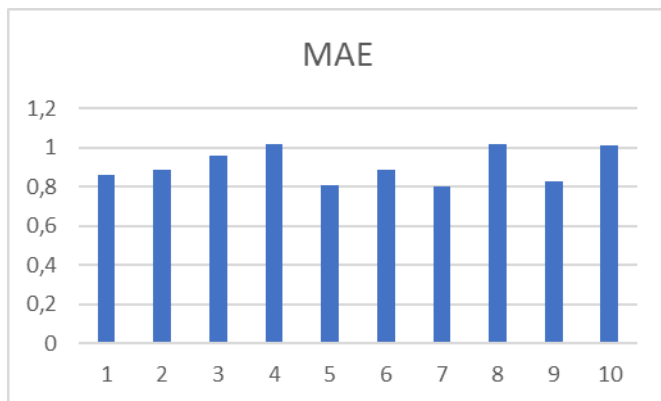
To check the performance of our recommendation system, we have used MAE(Mean Absolute Error) evaluation. 10 random users were taken from the test sub-dataset. Moreover, we have also chosen 10 random movies for each random user.

This metric can be calculated using the following equation 1:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - x_i| \quad (1)$$

Where  $n$  is the total number of values,  $x_i$  is the predicted rating, and  $y_i$  is the real rating.

Figure 2. provides a result for different experiments. We have calculated MAE 10 times on different variations of data.



(Figure 2) MAE values for 10 different experiments

It performed well with the MAE value of 0.94. We applied it to parallel processing independently of accuracy. Since it takes a very long time to detect a community, parallel processing should be applied to obtain better results in the future rather than sequential processing. In general, it is necessary to find a parallel processing method, create and apply a parallel model, but first of all, we would like to suggest the necessity of parallel processing through parallel programming.

And the execution time when we did not apply the parallel method was about 26 minutes.

On the other hand, after using parallelism for detecting communities we have gotten our result just after 13 minutes. Which is twice faster as the previous execution.

#### 5. Conclusion

In our paper, we have implemented a community detection algorithm Girvan–Newman to build a movie recommendation system. It performed well with the MAE value of 0.94. Additionally, the parallel programming method is also applied with the help of a *future* package to improve execution time.

In the future, we want to increase the size of the dataset and implement different methods for parallelism and concurrency.

#### References

- [1] Doo-Soon Park. Improved Movie Recommendation System based on Personal Propensity and Collaborative Filtering. KIPS transactions on computer and communication systems 2 (11), 475-482 (2013).
- [2] Phonexay Vilakone, Khamphaphone Xinchang, and Doo-Soon Park. Movie Recommendation System Based on Users' Personal Information and Movies Rated Using the Method of k-Clique and Normalized Discounted Cumulative Gain. J Inf Process Syst, Vol.16, No.2, pp.494~507, April 2020.
- [3] Vilakone, P., Park, DS., Xinchang, K. et al. An Efficient movie recommendation algorithm based on improved k-clique. Hum. Cent. Comput. Inf. Sci. 8, 38 (2018).
- [4] M. Newman, and M. Girvan, Finding and evaluating community structure in networks. Phys. Rev. E 69 (2): 026113 (February 2004)
- [5] M. E.J. Newman, Modularity and community structure in networks. Phys. Rev. E 64, 016132 (2001).
- [6] J. Diaz, C. Muñoz-Caro and A. Niño, "A Survey of Parallel Programming Models and Tools in the Multi and Many-Core Era," in IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 8, pp. 1369-1386, Aug. 2012.
- [7] Our World in Data, [https://ourworldindata.org/grapher/population-by-broad-age-group?country=~OWID\\_WRL](https://ourworldindata.org/grapher/population-by-broad-age-group?country=~OWID_WRL), last accessed 2023/04/05
- [8] International Labour Organization, <https://ilostat.ilo.org/resources/concepts-and-definitions/classification-occupation/>, last accessed 2023/04/05