ATLANTIS PRESS

*Advances in Computer Science Research, volume 87*

3rd International Conference on Mechatronics Engineering and Information Technology (ICMEIT 2019)

# Design and Implementation of Hardware Accelerator for Recommendation System based on Heterogeneous Computing Platform

Yang Li[1, 2, a], Zhitao Dai[1, 2, b]

[1]Beijing Key Laboratory of Intelligent Telecommunication Software and Multimedia, China.

[2]Beijing University of Posts and Telecommunication, Beijing, China.

[a] ly475652435@bupt.edu.cn, [b] daizt@bupt.edu.cn

**Abstract.** This study takes advantage of the combination of heterogeneous platform control and computing power, and optimizes the parallelization of the popular collaborative filtering recommendation algorithm. Compared with the traditional algorithm, the speedup has a certain degree of improvement and power consumption has also declined as well.

**Keyowrds:** heterogeneous computing platform, Recommendation System.

## 1. Introduction

The recommendation algorithm, as a classical algorithm of computer science, has been widely used in various fields. Various recommendation systems have been very popular in recent years and are used in various industries. Recommendations for the recommendation system include: movies, music, news, books, academic papers, search queries, categorization, and other products. The classification criteria of the recommendation algorithm are more diverse, and are generally divided into four categories, including collaborative filtering recommendation algorithm, content-based recommendation algorithm, hybrid recommendation algorithm and popularity recommendation algorithm, meanwhile, affected by the rapid development of today's machine learning field, in addition to the above four categories, it also includes model-based recommendation algorithms, recommendation algorithms based on deep learning, and so on. Among all the recommendation algorithm mentioned above, the currently popular and classic mature algorithm is the collaborative filtering recommendation algorithm [1]. The collaborative filtering recommendation algorithm can be divided into two types, one is the neighborhood model based algorithm and the other is the implicit semantic and matrix decomposition model based algorithms [2].

With the advent of the era of big data, the scale of data is growing rapidly. One of the most obvious features of the big data era is the sheer size of the data, and for the recommendation system, it is directly reflected in the number of new users and new items that are constantly flowing into the system and the increasing user behavior of the items and score records. Regardless of which recommendation algorithm, the ever-expanding data size makes the time spent in the training and prediction phases longer and longer, and the recommendation system has to take more time to generate recommendation information for the user. Therefore, in order to reduce the response time of the recommendation system and generate recommendation information for the user in time, it is necessary to speed up the execution of the recommendation algorithm.

At present, there are two mainstream acceleration platforms for algorithm acceleration: multi-core processor clusters and cloud computing platforms. The multi-core processor cluster consists of multiple compute nodes based on a general-purpose CPU. It mainly uses MPI, 0penMP, Pthread[3] and other programming models, with multi-process/multi-threaded approach to task-level/data-level parallelism. The cloud computing platform is also composed of many computing nodes based on general-purpose CPUs. It mainly uses Hadoop, Spark and other computing frameworks to perform task-level/data-level parallelism in the way of MapReduce [4].

The CPU (Central Processing Unit), which is a general-purpose processor, is an indispensable computing core in the computer. It performs a variety of calculation and processing tasks in daily work combined with the instruction set. However, in recent years, semiconductor technology

improvements have reached physical limits, circuits have become more complex and costly, and the traditional way of increasing computing power by increasing the CPU clock frequency and the number of cores has encountered heat dissipation and energy consumption bottlenecks. The demand for computing such as deep learning online prediction, video transcoding in live broadcast, image compression and decompression, and HTTPS encryption has far exceeded the capabilities of traditional CPU processors. The multi-core processor cluster and the cloud computing platform mentioned above as a whole have a good acceleration effect, but for a single computing node based on the general CPU architecture, the computational efficiency in processing the recommended algorithm task is relatively low, accompanied by High energy consumption as well.

At the same time, the co-processors such as GPU or FPGA and the traditional general-purpose processor CPU form a heterogeneous computing platform to achieve the improvement of computing performance, which has become a research hotspot in academia and industry [5]. As a special form of parallel computing, heterogeneous computing can assign different computing tasks according to the characteristics of each computing subsystem demonstrating the advantages that traditional architectures do not have in terms of improved algorithm performance, energy efficiency, and real-time computing. In the CPU-GPU heterogeneous architecture, most of the transistors in the GPU are used to fabricate the arithmetic unit, which has high computational power in computationally intensive, highly parallel, and simple control applications. These advantages are highly computationally efficient when dealing with recommended algorithm tasks, but the GPU's multiple power consumption of the CPU becomes an urgent problem for large-scale data centers that use GPUs as acceleration components. At the same time, due to the small cache size and low bandwidth of the GPU, the GPU computing core is often in a waiting state, which increases the data processing delay. In contrast, CPU-FPGA heterogeneous architecture has strong parallel processing capabilities, due to FPGA’s programmable dedicated processor composed of programmable logic blocks and interconnected networks [6], which can execute multiple threads under different logics to implement parallel processing pipelines. In high-performance computing applications, FPGA-specific logic circuits are directly executed by parallel computing hardware circuits, eliminating the need to follow the structure of the von Neumann stored program. Therefore, FPGAs have more powerful computing power and lower energy consumption than GPUs. Moreover, the FPGA has logical online reconfigurability and supports dynamic reconstruction of the algorithm [7]. The structure of the hardware circuit can be dynamically changed through software control, so that the system can balance the high performance of hardware calculation and the flexibility of software programming. The filtering algorithm can implement dynamic switching of each recommended application scenario.

## 2. Research Content

The main research content of this system is the collaborative accelerator recommendation algorithm hardware accelerator based on heterogeneous computing platform which is dedicated to accelerating traditional collaborative filtering recommendation algorithms. The first part is the related theory of collaborative filtering recommendation algorithms. The collaborative filtering recommendation algorithm is mainly divided into two categories: user-based collaborative filtering recommendation algorithm (User-based CF) and item-based collaborative filtering recommendation algorithm (Item-based CF). Their execution can be divided into two phases of training and prediction. User-based CF and Item-based CF mainly include the similarity/average difference calculation and the process of predicting the score based on the similarity value. Meanwhile, sorting the prediction scores when the TopN recommendation is required. Finding the computationally intensive parts of these two processes and passing the computationally intensive parts to the PL part of the heterogeneous processing chip through parallelization, pipeline, etc. is the key of this research. There are many metrics and methods for the concept of similarity. Generally, the similarity standards involved in this topic are as follows: Jaccard Coefficient , Euclidean. Distance, Cosine Similarity, etc.

For the selection of heterogeneous platforms used in this system, in the past few years, most researchers in the field of algorithm hardware acceleration and some large companies such as Intel,

etc., use the general-purpose computing graphics processor (GPU). Most of the research work on other programming frameworks such as OpenCL and OpenACC is relatively rare[8]. In recent years, the use of CPU-FPGA heterogeneous architecture has great room for expansion and mining potential for hardware acceleration. FPGA is a reconfigurable hardware structure that can flexibly change hardware for different applications. FPGA costs are relatively low. In terms of FPGA reconfigurable software development, HLS technology has matured in recent years compared with traditional RTL-level FPGA development. Xilinx's Vivado development kit and Altera's open, free standard OpenCL for writing executable programs for heterogeneous systems make it easier to design and develop FPGA software at higher levels in languages such as C. It seems that FPGA may be a better choice. In the choice of hardware platform, for example, the new FPGA device Zynq-7000 series chip proposed by Xilinx integrates high-performance ARM Cortex A9 hard core and off-chip programmable logic. In the Zynq platform, programmable logic can be thought of as a "peripheral" with reconfigurable features in the processor peripherals that can actively perform data interaction with external chips, or as a The master device that is peered with the processor. Through such a combination, the advantages of peripheral FPGA logic in parallel algorithm acceleration and dynamic reconfiguration are utilized, and the characteristics of the processor in processing complex control algorithms and operating the operating system are facilitated, thereby increasing flexibility and accelerating system.

The hardware accelerators referred to by the system mainly include training accelerators, prediction accelerators, and DMA devices. At the runtime, the controller notifies the DMA to initiate data transmission, and the hardware accelerator obtains data through DMA for corresponding calculation. For the training and prediction phases of the selected recommendation algorithm, the training accelerator and the prediction accelerator are respectively designed, which are essentially heterogeneous coprocessors to complete the assigned computing tasks under the control of the controller. The training accelerator is mainly composed of a control unit, a plurality of execution units and a DMA, and the control unit controls the execution unit, and the host CPU on the heterogeneous platform controls the control unit and the DMA; The predictive accelerator consists of multiple execution units and DMAs, each with its own controller. The overall structure of the training accelerator and predictive accelerator is as follows:

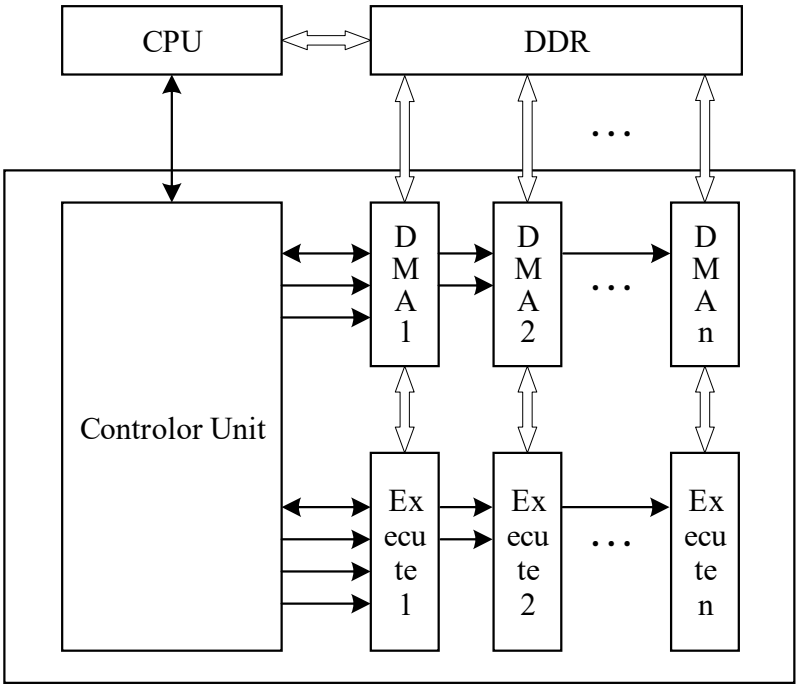


Fig. 1 Traing Accelerator

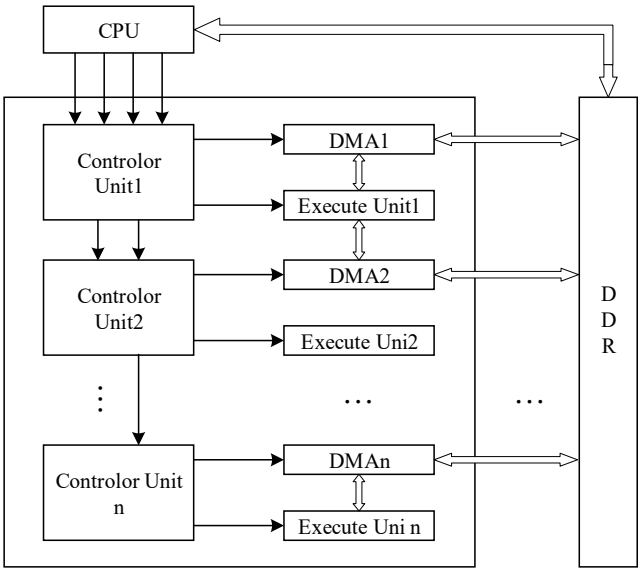


Fig. 2 Prediction Accelerator

## 3. Prototype Design

In summary, the goal of this study is to design and implement a set of accelerators for the training and prediction phases of the collaborative filtering recommendation algorithm on the heterogeneous computing platform. The hardware acceleration layer of the accelerator uses the heterogeneous computing of CPU-FPGA. The architecture combines the advantages of the host CPU control capability and the parallel computing power of the FPGA to accelerate the collaborative filtering recommendation algorithm. At the same time, combined with the dynamic reconfigurability of the FPGA mentioned above and the extensive use scenario of the collaborative filtering recommendation algorithm, the application scenario of the recommendation system can be avoided to achieve a wide range of effects.

In the process of in-depth study on the principle of user-based collaborative filtering recommendation algorithm and item-based collaborative filtering recommendation algorithm, the calculation hotspots of the training phase and prediction phase of collaborative filtering recommendation algorithm are analyzed. In the training phase, the user/item similarity matrix needs to be established based on the data in the dataset. In calculating the user/item similarity, the commonly used algorithms are Jaccard Coefficient , Euclidean Distance and Cosine Similarity. Whether it is a user-based or item-based collaborative filtering recommendation algorithm, only the x, y vector represents different things, the former represents the user and the latter represents the item. Combining the experimental results of the 100k dataset provided by MovieLens website on the CPU platform, it is concluded that in the training phase of collaborative filtering recommendation algorithm, the similarity calculation process takes the highest proportion of time, that is, the similarity calculation is the calculation hotspot of the training phase. See Table 1

Table 1. Calculation hotspot

| **algorithm** | ***Jaccard Coefficient*** | ***Cosine Similarity*** | ***Euclidean Distance*** |
|---|---|---|---|
| **Similarity calculation time ratio** | 97.89% | 97.9% | 98.02% |

According to the above analysis of the training phase algorithm of the collaborative filtering recommendation algorithm, the training accelerator is responsible for accelerating the similarity of each pair of user/item vectors in the user/item, and each pair of vectors has no dependency when calculating the similarity, so multiple executions can be used. The units are calculated in parallel, and all the computing units share the entire computing task together, which can greatly simplify the hardware control logic design while ensuring the operating efficiency of each unit.

In the prediction phase, according to the type of task, it can be divided into a score prediction task and a TopN recommended task. For the scoring prediction task, since each user/item to be predicted

is independent of each other and there is no dependency, multiple execution units can be used for parallel calculation. In addition, considering that the neighborhood of each user/item is different, that is, the number of numerators and denominators of Equation 3-4 is different, which results in different execution cycles per execution unit, so each execution unit of the prediction accelerator has its own The instruction cache is not interfered with each other during execution, avoiding the overhead of hardware unit synchronization. For the TopN recommended task, the result of the prediction score needs to be transmitted back to the PS part through DMA, because the sorting work can be parallelized into not high, and the processing capability of the ARM chip of the PS part is sufficient, so the final sorting recommendation task is handed over to the host CPU.

## 4. Conclusion

Using the designed training accelerator and predictive accelerator prototype, combined with the MovieLens data set, the acceleration effect of the recommended algorithm accelerator is simulated and verified. According to the number of clock cycles running on the ARM CPU, Core i3 CPU and training accelerator of the ml-100k data set, the training accelerator acceleration ratio shown in the figure and the prediction accelerator acceleration ratio shown in the figure are obtained.

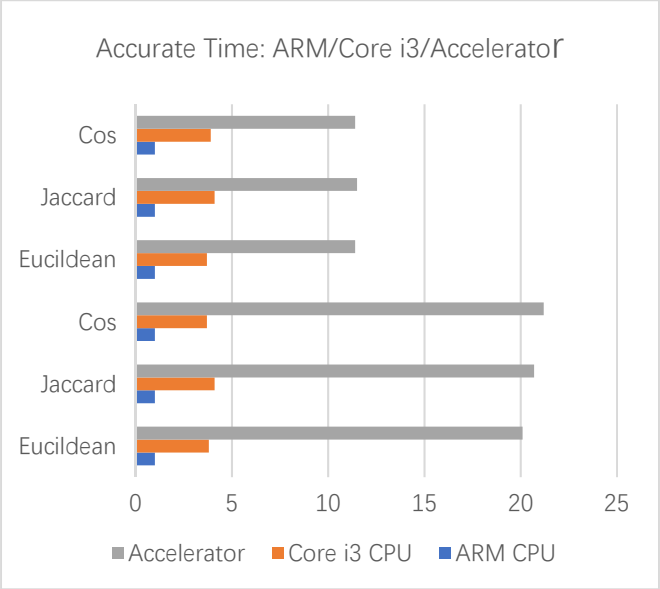


Fig. 3 Training Accelerator Speed-Up

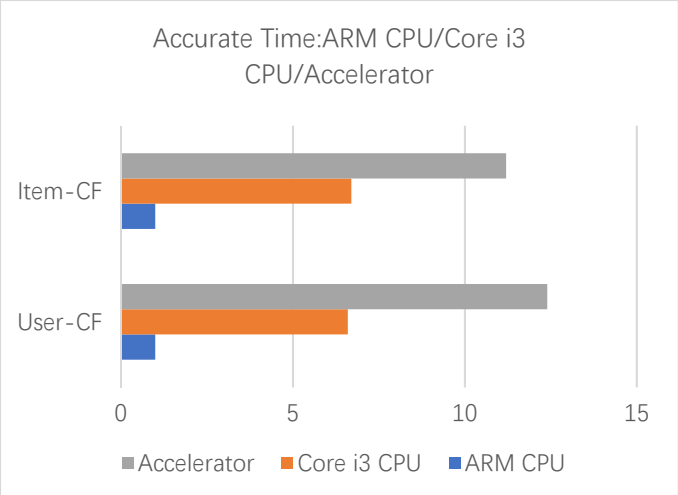


Fig. 4 Training Accelerator Speed-Up

It can be seen from the figure that the acceleration ratio of the training accelerator to the ARM CPU is about 20 when performing the user-based collaborative filtering recommendation algorithm, and the acceleration ratio of the Core i3 CPU is about 4; the training is performed when performing the item-based collaborative filtering recommendation algorithm. Accelerator vs. ARM CPU has an acceleration ratio of around 11 and the Core i3 CPU has an acceleration ratio of around 2. In addition to the comparison of the speedup ratio, the FPGA has a unique advantage over the traditional CPU platform in terms of power consumption ratio. Therefore, the heterogeneous accelerator has a good acceleration effect on the collaborative filtering recommendation algorithm and has a lower energy consumption.

## References

[1]. Collaborative_filtering (CF): https://en.wikipedia.org/wiki/Collaborative_filtering.

[2]. Stevens W R, Rago S A. Advanced programming in the UNIX environment [M]. AddisonWesley, 2013.

[3]. White T. Hadoop: The definitive guide [M]. O'Reilly Media, Inc, 2012.

[4]. Narang A, Srivastava A, Katta N P K.High performance offine and online distributed collaborative filtering [C]/ Data Mining (ICDM), 2012 IEEE 12th International Conference on. IEEEE, 2012: 549-558.

[5]. Zhanchun G, Yuying L. Improving the collaborative filtering recommender system by using gpu [C]// Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2012 International Conference on. IEEE, 2012:330-333.

[6]. ABDELFATTAH M S, HAGIESCU A, SINGH D. Gzip on a chip: high performance lossless data compression on FGPAs using OpenCL [C]// International Workshop on Opencl, 2014:1-9.

[7]. Yu Q, Wang C, Ma XZ, et al.A Deep Learning prediction process accelerator based FPGA [C]// Cluster, Cloud and Grid Computing (CCGrid), 2015 15th IEEE/ ACM International Symposium on. IEEE, 2015: 1159-1162.

[8]. Yi Shan, Bo Wang FPMR: MapReduce framework on FPGA, Proceedings of the 18th annual ACM/SIGDA international symposium 2010.