

Research on Intelligentization of Cloud Computing Programs Based on Self-awareness

Hanpeng Liu

School of Computer Science & Engineering
Xi'an Technological University
Xi'an, China
E-mail: 1226594381@qq.com

Junmin Luo

School of Computer Science & Engineering
Xi'an Technological University
Xi'an, China

Wuqi Gao

School of Computer Science & Engineering
Xi'an Technological University
Xi'an, China
E-mail: gaowuqi@126.com

Abstract—Through the research of MapReduce programming framework of cloud computing, the current MapReduce program only solves specific problems, and there is no design experience or design feature summary of MapReduce program, let alone formal description and experience inheritance and application of knowledge base. In order to solve the problem of intelligent cloud computing program, a general MapReduce program generation method is designed. This paper proposes the architecture of intelligent cloud computing by studying AORBCO model and combining cloud computing technology. According to the behavior control mechanism in AORBCO model, a program generation method of MapReduce in intelligent cloud computing is proposed. This method will extract entity information in input data set and entity information in knowledge base in intelligent cloud computing for similarity calculation, and extract the entity in the top order as key key-value pair information in intelligent cloud computing judgment data set. The data processing types are divided, and then aligned with each specific MapReduce capability, and the MapReduce program generation experiment is verified in the AORBCO model development platform. The experiment shows that the complexity of big data MapReduce program code is simplified, and the generated code execution efficiency is good.

Keywords-Cloud Computing; Artificial Intelligence; Intelligence; MapReduce; AORBCO Model

I. INTRODUCTION

At present, cloud computing is not only a kind of distributed computing, but also the result of the mixed evolution and leap of distributed computing, utility computing, load balancing, parallel computing, network storage and virtualization. Because cloud computing is the result of mixed evolution of various technologies, its maturity is high, and it is promoted by large companies, and its development is extremely rapid. MapReduce is the main programming model of cloud computing, which is used to process and generate large datasets for various tasks in the real world [2]. Dayanand and others put forward an optimized HPMR (Hadoop MapReduce) model, which balances the performance between I/O system and CPU [3]. Liu Jun and others proposed a configuration parameter adjustment method based on feature selection algorithm, which improved the working efficiency of MapReduce in Hadoop [4]. Abolfazl Gandom et al. proposed a heterogeneous cluster job execution time prediction model based on MapReduce stage. In addition, a novel heuristic method is designed to enhance the performance of MapReduce clusters and reduce their job execution time [5]. At present, the main code generation methods include

template-based program generation, which generates source code according to input and template file based on certain code generation engine [6]. The automatic code generation method based on document comment parsing generates source programs with similar functions for comments in the software source code, and uses the code generation engine to convert the comments into corresponding codes [7].

Although cloud computing offers different services and focuses on a variety of data-processing applications, the field of cloud computing is not closely associated with human intelligence. Automatic program generation is the core of artificial intelligence. Program intelligence is to model human intelligence and simulate human problem-solving mechanism [8]. At present, cloud computing technology still completely relies on human beings to complete the development of corresponding services, and at the computing level, it completely relies on human beings to write corresponding MapReduce computing programs according to the requirements, and lacks the formal description of knowledge representation of MapReduce program cases and the summary and application of cloud computing programming knowledge.

In this paper, we believe that the study of the architecture of intelligent cloud computing should first study the nature and characteristics of human intelligence. On the basis of AORBCO (Agent-Object-Relationship Model Based on Consistency-Only), an intelligent cloud computing architecture is established, and the advantages of human intelligence are simulated by AORBCO model. Integrating the intelligent cloud computing architecture based on self-awareness into MapReduce, a programming framework in cloud computing, and combining with the four characteristics of human intelligence, provides a general scheme for the intelligent realization of cloud computing programs.

II. INTELLIGENT CLOUD'S DEFINITION BASED ON SELF-CONSCIOUSNESS

At present, the industry has not formed a unified view on the definition of cloud computing. Liu Peng, an expert in grid computing and cloud

computing in China, gives the following definition: "Cloud computing distributes computing tasks on a resource pool composed of a large number of computers, so that various application systems can obtain computing power, storage space and various software services as needed" [9]. It is called "cloud" because it has the characteristics of real clouds in some aspects: clouds are generally large; The scale of the cloud can be dynamically scaled, and its boundaries are blurred. Cloud computing is the integration of all desired information services, just like the aggregation of "clouds" to unite the power of the Internet. As long as users have needs, they can use the equipment to quickly find the required services at any time and any place.

Cloud computing technology itself can fuse data from different computers. If cloud computing can introduce human intelligence, cloud computing can bring new intelligence. Through the study of human intelligence in general psychology, reflective psychology, cognitive psychology and cognitive psychology [10-13], it is found that cognitive psychology has thoroughly studied the essence, composition and function of human intelligence, and summed up four characteristics of human intelligence, such as self-awareness, mutual expressiveness, fuzziness and dynamics, as shown in Figure 1.

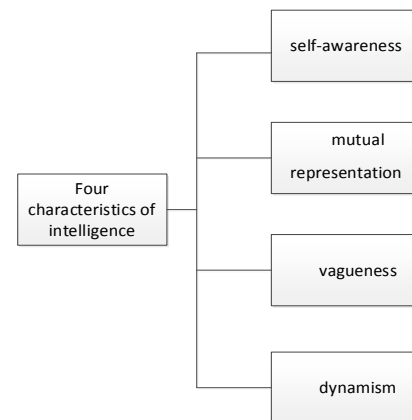


Figure 1. Four characteristics of intelligence

Similarly, like intelligence, intelligent cloud has four characteristics. How to make cloud computing simulate human intelligence is an important research issue of intelligent cloud computing. To make cloud computing intelligent,

we must first understand the meaning and characteristics of intelligence. As shown in Figure 1, self-consciousness means that individuals know their own state, emotion and thinking, and know their acquaintances through interaction with acquaintances. Everyone also has an independent world. Mutual expressiveness means that everything in the real world depends on other things and is not completely isolated. Fuzziness means that the relationship between things has the difference between closeness and distance, and it is not a clear classification. Dynamism means that the state and relationship of things change with the change of environment.

In human society, cooperation between human beings is more intelligent and efficient than single processing. There are all kinds of knowledge in human society, and cloud society can be regarded as a centralized society that brings human knowledge together, and cloud society will eventually be more intelligent than human society. Cloud computing connects the real world with the virtual world (virtual resources), while intelligent cloud connects the real world with the abstract world. AORBCO model is a knowledge-based model, and cloud computing provides services in the form of services. The concept of cloud is sharing, and the concept of computing is to provide solutions to problems. Therefore, the AORBCO model is considered to be a intelligent cloud and a solution service model based on general problems. Acquaintances are the concrete service resources in intelligent cloud computing. Desire decomposition in AORBCO model is also a distributed way to solve problems, and AORBCO model is a natural distributed system. From the perspective of AORBCO model, cloud computing reflects the corresponding ability in the model. Cloud computing is a service form in the model, which provides services to the real world through cloud computing.

III. INTELLIGENT CLOUD COMPUTING ARCHITECTURE AND ITS IMPLEMENTATION MECHANISM

A. Intelligent cloud computing architecture

Cloud computing technology itself has an architecture, which summarizes the main features

of different solutions. According to the related technologies and services provided by cloud computing, the architecture of cloud computing technology is divided into four layers: physical resource layer, resource pool layer, management middleware layer and SOA (service-oriented architecture) construction layer. However, each solution in the architecture of cloud computing may only realize some of its functions, and some relatively minor functions have not been summarized, so intelligent cloud computing is proposed. The architecture of intelligent cloud computing consists of six parts: belief, ability, desire, planning, execution and behavior control mechanism. Among them, belief represents the description of recognized entities and their relationships by Ego, entities are acquaintance sets and objects sets, and relationships are class sets. Belief also represents descriptive knowledge; Ability represents the operation that Ego can perform, that is, process knowledge; The intelligentization of cloud computing programs, as a capability of Ego, provides an intelligent program service to the real world. Desire is a set of states that Ego wants to achieve, and it is a description of the desired state that Ego produces after perceiving and understanding information. Planning is an Ego's plan to solve problems by combining strategic knowledge (special process knowledge) in order to fulfill its current wishes. Execution means the execution of the planning scheme; The behavior control mechanism is the "controller" of Ego, which is responsible for the coordination of various modules and the intelligent operation of Ego. The architecture of intelligent cloud computing is shown in Figure 2.

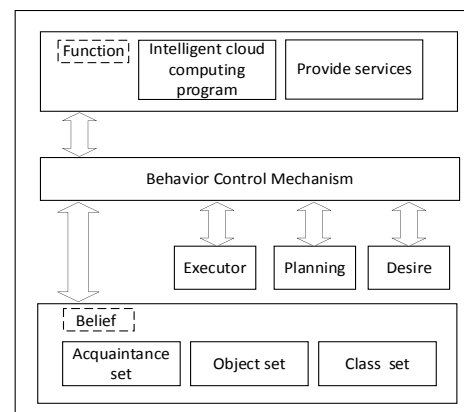


Figure 2. Intelligent Cloud Computing Architecture Diagram

All the subjects in the cloud computing cluster, that is, the Agent in the cloud computing field, abstract all the subjects known by Ego. Intelligent cloud cluster usually adopts a master-slave structure, with Ego as the manager of the cluster, while acquaintances in the cluster are ordinary members. The difference between them lies in the different execution rights and tasks within the cluster. AORBCO model divides Agents into Ego and Acq_Agent. There is an equal relationship between acquaintances recognized by Ego, that is, acquaintances in Ego's cognition are at the same level, and there is also an unequal relationship between superiors and subordinates, that is, acquaintances in Ego's cognition are at different levels. In the cluster of intelligent cloud computing, Ego is the global manager and the system management and task allocation of the whole cluster. All the subjects in the cloud computing cluster, that is, the Agent in the cloud computing field, abstract all the subjects known by Ego. Intelligent cloud cluster usually adopts a master-slave structure, with Ego as the manager of the cluster, while acquaintances in the cluster are ordinary members. The difference between them lies in the different execution rights and tasks within the cluster. AORBCO model divides Agents into Ego and Acq_Agent. There is an equal relationship between acquaintances recognized by Ego, that is, acquaintances in Ego's cognition are at the same level, and there is also an unequal relationship between superiors and subordinates, that is, acquaintances in Ego's cognition are at different levels. In the cluster of intelligent cloud computing, Ego is the global manager and the system management and task allocation of the whole cluster.

The detailed task processing process of intelligent cloud computing cluster is as follows: the client submits a task requirement to intelligent cloud cluster. After the task is submitted, if the requirements of the task are solved by desire decomposition by Ego, the sub-wishes decomposed by Ego for different wishes will be different and the sub-wishes will match different abilities in the Ego competence library; If the requirements of tasks need to be solved by the desire decomposition of Ego and acquaintances,

Ego is the manager of the whole cluster, and acquaintances are responsible for data processing of tasks. Because acquaintances have different abilities, each acquaintance has different degrees of decomposition of wishes, and the degree of decomposition granularity of wishes depends on acquaintances' computing ability. According to the intimacy with acquaintances and Ego's cognition of acquaintances' ability, Ego assigns specific tasks to acquaintances in the cluster after wish decomposition, selects some acquaintances to generate application planners for each specific task, submits task planning applications to Ego, manages each specific task and reports the task implementation to Ego. Other acquaintances who perform specific computing tasks will call MapReduce capabilities through the capability library to perform computing tasks. According to the implementation of each small task and their own state information, they are provided to acquaintances with application planners. The architecture of intelligent cloud computing cluster is shown in Figure 3.

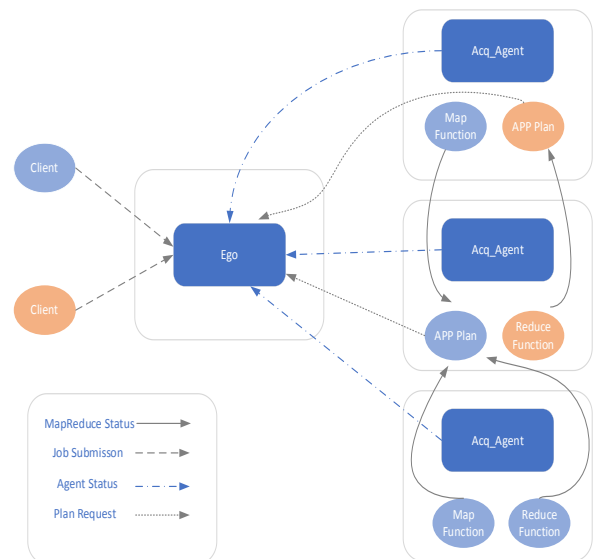


Figure 3. Structure diagram of intelligent cloud computing cluster

B. Implementation mechanism of intelligent cloud computing

The implementation mechanism in the intelligent cloud computing architecture is modeled around Ego. The model divides the process of Ego understanding the world or solving problems into five steps: perception,

understanding, planning, execution and learning. In the AORBCO model, the research on the intelligence of cloud computing programs is to establish contact with Ego's beliefs, abilities and wishes after Ego perceives the problem demand of the external world, so as to generate a plan for generating MapReduce programs for solving problems and realize the generation of cloud computing programs. Execute an operation that represents the action generated for the plan; Learning refers to the experience summary of Ego, including unintentional learning and intentional learning. Unintentional learning refers to the habitual change of ability proficiency when Ego knows the world. Intentional learning means that when Ego encounters new problems, it learns new abilities from acquaintances or solves new problems through its own planning. Based on the architecture of intelligent cloud computing, the implementation mechanism of intelligent cloud computing is shown in Figure 4.

IV. MAPREDUCE PROGRAM GENERATION IN INTELLIGENT CLOUD

In the intelligent cloud model with Ego as the core, a network structure is formed among acquaintances in Ego beliefs. Ego needs to consult and plan unknown problems by requesting acquaintances. In the process of providing services in the intelligent cloud, the desire is decomposed according to the process knowledge. Although the model breaks down desires, it is the perceived external environment goal itself that needs to be broken down when the actual problem is solved. Intelligent cloud computing reflects a distributed way to deal with problems. In an intelligent cloud computing system, Ego learns new abilities by interacting with acquaintances and solving problems together, in addition to the basic abilities given during initialization, when perceiving the external environment and dealing with perceived problems. MapReduce program generation in intelligent cloud is the key to realize intelligent cloud computing program.

The intelligent process of cloud computing program first needs to define complete subtasks in the intelligent process of cloud computing, that is, data type analysis and MapReduce feature module matching algorithm. In this paper, the cloud computing program intelligence (program generation) in AORBCO model is defined as $PG=\{I,E,C,P,J\}$. I is input information, which can be input data set document or natural language text; E is entity, parsing input information into key entity information, the main part of data type analysis and processing. C is the constraint information of the parameter. P is the parameter set, which mainly includes the type and number of parameters; J indicates a MapReduce job.

The MapReduce program generates a detailed definition process as follows: The input information can be a data set document or natural language text. In this step, the input information needs to be preprocessed and analyzed. According to the input information that has been extracted, the data processing type is analyzed, and the input information is converted into key entity information. Through the key entity information, the specific requirements and desires of the input information and the data processing type of the

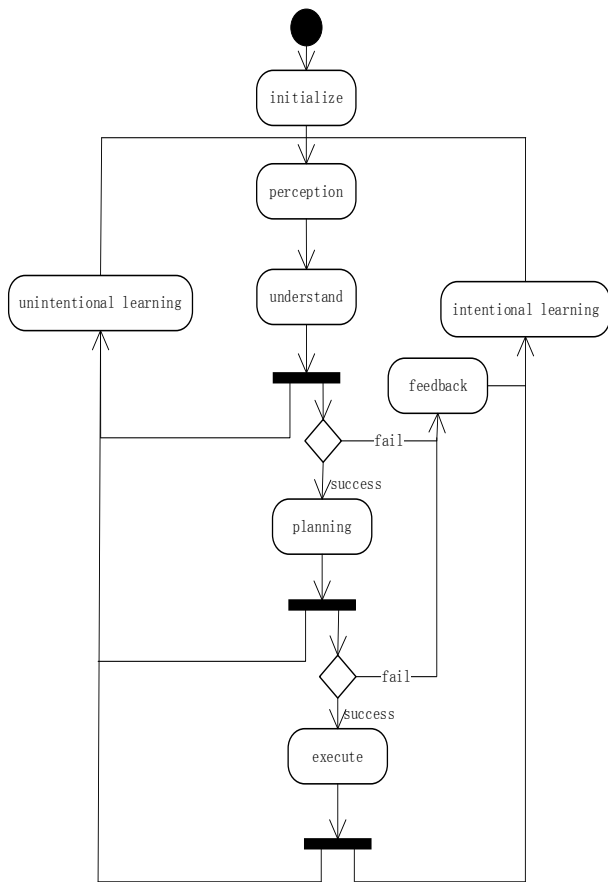


Figure 4. Activity Diagram of Intelligent Cloud Computing Implementation Mechanism

input requirements are determined. According to the analysis results of data processing types and actual requirements, the constraint conditions of program parameters are determined. The root constraint and entity information determine the program's set of parameters, including the type and number of parameters. Finally, MapReduce programs, including Mapper and Reducer functions, are generated by ability alignment to improve the intelligence level of cloud computing programs and achieve fast and efficient data processing and analysis.

A MapReduce program in cloud computing can be described as a seven-tuple: input data set is, where each data record contains key-value pairs (k_i, v_i) . Then the MapReduce task program can be represented as a seven-tuple $j = \langle I, M, C, P, S, R, O \rangle$.

I is InputFormat specifies the format of the input data, $I: D \rightarrow k_1 \times v_1$, which maps each data record in D to a key-value pair, which maps each data record in D to a key-value pair (k_1, v_1) .

M is the Map function, $M: (k_1 \times v_1) \rightarrow k'_2 \times v'_2$, which takes one key-value pair (k_i, v_i) and converts it into several new key-value pairs k'_j, v'_j .

C is the Combine function. $C: (k'_2, [v'_2]) \rightarrow (k'_2, v'_2)$, performs a local combine operation on the key-value pairs output by the Map function to Reduce the scale of input data and the overhead of network transmission.

P is the Partition function, $P: k'_2 \rightarrow [0, N - 1]$, which divides key-value pairs $(k'_j \times v'_j)$ into N different partitions. The rules of the partitions can be specified explicitly by the user or set by the framework default.

S is the Sort function, $S: (k_2, v_2) \rightarrow (k_2, v_2)$, which sorts the key-value pairs in each partition by its own key.

R is the Reduce function, $R: (k_2, [v_2]) \rightarrow (k_3, v_3)$, it will be the same key value list for the convention operation, the description of the new key-value pair list $[(k_1, v_1)]$.

O is the OutputFormat output format, $O: [(k_3, v_3)] \rightarrow Y$, which converts the result

set $[(k_1, v_1)]$ to the format specified by OutputFormat for output.

A user-defined MapReduce program can be clearly defined through the above seven tuples, with clear definitions from input data to output results. Among the components of MapReduce computing task, M and R parts in the program are the core of MapReduce program and must be possessed, while C, P, S and S can be decided by users whether to use the default functions of MapReduce framework.

A. Analysis of data processing types

Secondly, in order to realize the intelligentization of cloud computing program, it is necessary to divide the business logic of big data processing into different types, so as to carry out targeted program intelligentization for each type. It is also necessary to determine the type of task requirements they belong to before a particular MapReduce program is built. For the document requirements with significant text feature information, then according to the keyword matching text feature class, the input document requirements are automatically determined. Match the MapReduce program based on the document requirements. The cosine similarity can be used to calculate the text content and the prior knowledge in the Ego knowledge base for similarity calculation, so as to judge the actual demand of the document and provide intelligent matching MapReduce program code according to the demand.

TextRank algorithm is to convert a document into a directed weighted word graph model, which divides the text into basic units, namely words. Each basic unit is regarded as a node, and the edge between each node is determined by the co-occurrence relationship between word nodes, while the importance of nodes is determined by the number of pointing adjacent nodes. Construct TextRank keyword graph $G = (V, E)$, where V is node set and E is edge set between nodes. TextRank algorithm is calculated as follows.

$$WS(V_i) = (1 - d) + d \times \sum_{V_j \in in(V_i)} \frac{w_{ji}}{\sum_{V_k \in out(V_j)} w_{jk}} WS(V_j) \quad (1)$$

Which $\ln(v_i)$ represents the node set pointing to the node v_i ; $\text{out}(v_j)$ represents the set of nodes pointed by v_j , W_{ji} represents the weight from node W_j to W_i -side. d is the damping coefficient, which represents the probability of transferring from any designated node in the graph to other nodes, and its value range is $[0,1]$. If the damping coefficient is too large, the number of iterations will increase sharply and the ranking of the algorithm will be unstable; If the damping coefficient is too small, the iterative process has no obvious effect, which ensures that the weight can be stably transferred to convergence. Finally, the weight of each word is calculated and sorted, and generally it is 0.85 [14].

TextRank algorithm extracts keywords as key entity information, and judges the similarity of the class to which the entity belongs. Sim A and B represent the similarity of two entities A and B , where A is the key entity extracted from the text, B is the entity in the Ego knowledge base, and R represents the prediction result. When R is $+1$, it indicates that the two entities are related, and the relationship can be judged. Each dimension of the vector represents the degree to which the entity belongs to the I -th class. By calculating the cosine similarity between two vectors, the similarity of the class is obtained. Cosine similarity is a measure of the similarity of two vectors in the direction. The calculation process of cosine similarity is as follows:

$$\text{Sim}(A, B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^n (A_i \times B_i)}{\sqrt{\sum_{i=1}^n A_i^2} \times \sqrt{\sum_{i=1}^n B_i^2}} \quad (2)$$

In the formula, a and b are two input vectors respectively, and the value range of is $[-1,1]$.

The formula of the $\text{sign}(x)$ function is shown as follows:

$$R = \text{sign}(\text{Sim}(A, B)) \quad (3)$$

$$\text{sign}(x) = \begin{cases} -1, & x < 0 \\ +1, & x \geq 0 \end{cases} \quad (4)$$

If you can't find this type of document requirements in the knowledge base of Ego, you can enhance the semantic recognition of requirements by user input and natural language processing. In natural language, the key information in the sentence is extracted to the triple information in natural language, and Ego uses the same method to calculate and judge the customer's requirements, thus giving a MapReduce program for this requirement.

B. Matching of MapReduce module

The matching of MapReduce module features is closely related to the capability alignment in AORBCO model, and the formal definition of capability Function in AORBCO model is shown in the formula:

$$F(t) = \{name, parameter, precondition, postcondition, body\} \quad (5)$$

Where $name$ represents the name of the ability and $parameter$ represents the parameters of the ability, including input parameters and output parameters (return value). Preconditions and postconditions are constraints on parameters, where preconditions represent preconditions and postconditions represent postconditions. Body is the ability body, which represents a series of operations from the initial state to the target state of ability execution. The constraints of preconditions are the types and numbers of parameters and the judgment of the class membership degree of Ego's requirements for text documents, and these constraints are expressed by logical expressions. The constraints of postconditions are to judge the parameter types and numbers of the corresponding capabilities of Ego, and these constraints are expressed by logical expressions. According to the number and type of parameters and the description of preconditions and postconditions, the logical expressions of parameters are merged through the conjunction and disjunction relationship. If the corresponding capability description file can be matched in the capability library of Ego, then the corresponding MapReduce program can be matched according to the module feature template of the capability. If the corresponding capability is not matched from the capability library, similar MapReduce

programs can be recommended through fuzzy matching.

Based on the data processing tasks applicable to MapReduce program, the corresponding Map and Reduce function operations and parameters are summarized, and the corresponding MapReduce programming module features are formed. The Map function takes KEYIN1 and VALUEIN1 as

input parameter types, and the Context class object completes the writing of output content, then the reduce function indicates the types of keys and values with KEYOUT2 and VALUEOUT2 respectively, and the types of input keys and values are the same as the output types of the Map function. Figure 5 shows the module feature diagram of MapReduce programming model.

```
public static class Map extends Mapper < KEYIN1, VALUEIN1,
KEYOUT1, VALUEOUT1 > {
    global variable declaration;
    public void map (KEYIN1 K1, VALUEIN1 V1, Context context) throws
IOException, InterruptedException {
        local variable declaration;
        format conversion;
        data processing operations;
        context.write(KEYOUT1, k2, VALUEOUT1 v2); }
}
public static class Reduce extends Reducer < KEYOUT1,
VALUEOUT1, KEYOUT2, VALUEOUT2 > {
    global variable declaration;
    public void reduce (KEYOUT1 K2, Iterable < VALUEOUT1 >
values, Context context) throws IOException, InterruptedException {
        local variable declaration;
        for(VALUEOUT1 Val: values){
            merge operation; }
        context.write(KEYOUT2 k3, VALUEOUT2 v3); }
}
```

Figure 5. Module Features of MapReduce Programming Model

Finally, the feature selection of MapReduce function and Reduce function is matched by rule based method. Production rules should be the embodiment of strategic knowledge application in intelligent model theory. In intelligent MapReduce, users' demands are received through the perception module in the behavior control mechanism of AORBCO model. The selection according to the characteristics of Map function is understanding, and the planning and execution of Map, Shuffle and Reduce processes are planning and execution. The MapReduce program in the intelligent cloud determines the data processing type based on the text content and matches corresponding rules. After matching different rules, different Map functions are selected for processing. The Map function processing is handed over to the Reduce function for processing and the final processing

result is returned. According to input and output type information and code template, the behavior control mechanism in AORBCO model is used to generate the corresponding code. Code templates include Job configuration templates, Mapper templates, Reduce templates, and Key/Value templates related to the MapReduce execution platform. The model ultimately generates MapReduce code based on user-provided input.

C. Experimental verification

On the premise that Ego owns various MapReduce programs, the priori knowledge and behavior control mechanism of AORBCO model are used to analyze the data processing types of text contents, determine the class of contents, and then find the MapReduce program code that can be solved from Ego's ability through MapReduce

module feature matching. This experiment verifies that the average score is obtained by inputting the text document, and Ego senses the content through prior knowledge. The background perception input and results are shown in the figure respectively. Since this experiment mainly verifies the

feasibility of MapReduce program generation in intelligent cloud on AORBCO model platform, it does not involve communication between Ego and acquaintances, asking acquaintances for assistance, etc., and Ego is used to solve problems independently by default.



Figure 6. AORBCO model development platform

In the MapReduce program generated by AORBCO model development platform, the benchmark program WordCount is selected as the experiment case. The experimental data set is a random text data generated by Hadoop RandomTextWriter. In this experiment, the task completion time of word frequency statistics was analyzed by comparing MapReduce program and Java program under different data set document sizes. As shown in Figure 7, experimental tests show that Java programs perform better with smaller input document data sizes. As the size of the input document data increases, the MapReduce program performs better than the Java program, reflecting the advantages of MapReduce program distributed computing.

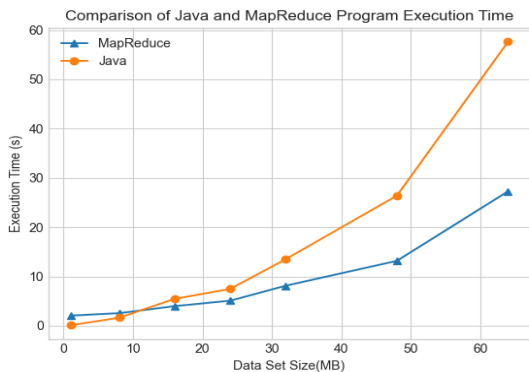


Figure 7. Experimental result diagram

V. CONCLUSIONS

Based on the current situation that the application of big data is more and more extensive, and the cloud computing MapReduce program only solves specific problems. This paper expounds the definition of intelligent cloud on the basis of self-awareness. Based on the architecture of AORBCO model, the architecture of intelligent cloud computing technology is given, and the generation method of MapReduce program is proposed in intelligent cloud computing architecture. The experimental results show that the MapReduce program can deal with the data processing task of cloud computing under large data scale more generally, and the availability of the MapReduce program generation method is verified by an example, and the workload of developing the MapReduce program is reduced. The subsequent research is based on the intelligent research of cloud computing program in the AORBCO model, that is, the MapReduce program is improved into a more general cloud computing program under the intelligent cloud computing architecture.

REFERENCES

[1] Wu Xing, Wang Minchao, Zhang Wu, Li Qing. Review of cloud computing research [J]. Science & Technology Innovation and Productivity, 2011(06):49-55.

- [2] Wang Jia. MapReduce Job Scheduling in distributed Heterogeneous Cluster [D]. Southeast University, 2019.
- [3] Sontakke V, Dayanand R B. Optimization of hadoop mapreduce model in cloud computing environment[C]//2019 International Conference on Smart Systems and Inventive Technology (ICSSIT). IEEE, 2019: 510-515.
- [4] Liu Jun, Tang Sule, Xu Guangxia, Ma Chuang, Lin Mingwei. A Novel Configuration Tuning Method Based on Feature Selection for Hadoop MapReduce [J]. IEEE Access, 2020, 8.
- [5] Abolfazl Gandomi, Ali Movaghar, Midia Reshadi, Ahmad Khademzadeh. Designing a MapReduce performance model in distributed heterogeneous platforms based on benchmarking approach [J]. The Journal of Supercomputing, 2020, 76(9).
- [6] Wang Bo, HUA Qingyi, Shu Xinfeng. An automatic code generation method based on Model and Template Fusion [J]. Modern Electronics Technique, 2019, 42(22):69-74.
- [7] Lin Zeqi, Zou Yanzhen, Zhao Junfeng, Cao Yingkui, Xie Bing. Semantic Search method of software Documents based on code structure knowledge [J]. Journal of Software, 2019, 30(12):3714-3729.
- [8] Feng Yanxing. Research on Program Generation in AORBCO Model [D]. Xi'an Technological University, 2021.
- [9] Liu Peng. Definition and Characteristics of cloud computing [EB/OL]. [2009-02-15]. <http://www.chinacloud.cn/>.
- [10] Luo Junmin, Wang Lei. Research on dynamic ontology architecture based on mutual tabulation [J]. Microelectronics & Computer, 2013, 30(2):4.
- [11] Luo Junmin, Wu Yuyun, Wu Bin. Research on fuzzy ontology and its Evolution [J]. Microelectronics & Computer, 2011, 28(5):4.
- [12] Luo Junmin, Li Junwei. Research on Agent Self-awareness [J]. Microelectronics and Computers, 2015.
- [13] Gong Min, Luo Junmin, Gao Wuqi. Intelligent model description language based on self-awareness. Science Technology and Engineering, 2018, 18(31):6.
- [14] Huang Bo, LIU Chuan-cai. Chinese Automatic Text Summarization Based on Weighted TextRank [J]. Application Research of Computer, 20, 37(02):407-410.