

Clustering Integration and Incremental Learning Model under Spark on Dynamic PPI Data

Dafeng Gong

Department of Information Technology, Wenzhou Vocational & Technical College, Wenzhou,

China

289133894@qq.com

Abstract: In order to improve the performance of the group intelligence algorithm, the parallelism of the group intelligence algorithm and the parallel computing power of the Spark framework were applied. The cluster integration and incremental learning models under Spark were used to obtain the results of the original PPI network. Through the refinement operation, the local optimum of the clustering results was effectively avoided. The results showed that compared with other detection algorithms, the algorithm obtained better results and efficiently processed large-scale data. In summary, the model has potential application value. The results have expanded the research and application of group intelligence and related technologies to a certain extent. The actual engineering optimization problem is effectively solved.

Keywords: Group intelligence algorithm, Spark, PPI data, Clustering integration.

1. INTRODUCTION

Life sciences offer exciting opportunities. The completion of the Human Genome Project is a major step in the development of life sciences. Biological research has entered the post-gene era. In the post-gene era, one of the most important research priorities in the biological sciences is proteomics. Studies have found that proteins do not act alone in the biochemical process of a cell, but through the interaction of proteins and proteins to form a macromolecular complex to complete a biological function. Therefore, the interaction between proteins is the basis for understanding the organization, processes, and functions of cells in life. In the field of biology, the study of protein-protein interactions can not only understand biological processes from a systematic perspective and reveal the mechanisms of disease, but also help people find new drug targets. It plays a positive role in the development and design of new drugs. Therefore, in the post-gene era, the study of protein-protein interaction (PPI) has become one of the important research fields of proteomics. It has received extensive attention from scholars at home and abroad.

In recent years, with the emergence and development of high-throughput biological experimental methods and techniques such as protein chip and yeast two-hybrid and the widespread application of

data mining technology in PPI prediction, the number of available PPI data has increased dramatically to form a PPI network. An important task in the post-gene era is to study the biological implications of a large amount of biological data. Therefore, the rapid and efficient identification of functional modules with biological functions in a large number of available PPI networks has become one of the most critical issues in proteomics research.

Biologists use traditional and classic biological experiments to detect PPI network function modules. However, due to the limitations of these detection methods in terms of cost, time and quality of detection, the actual needs of human life science research in the post-gene era cannot be met. Biology experts have discovered that the tightly linked regions of the protein present in the PPI network generally correspond to functional modules of the protein. Therefore, based on machine learning and data mining, the rapid development of methods for detecting functional modules by identifying the tight connection structure of the network has become an effective way to detect functional modules in PPI networks.

2. STATE OF THE ART

As an emerging intelligent computing technology, group intelligent algorithm has attracted the attention of many scholars. Inspired by many adaptive optimization phenomena in nature, the bio-population mechanism was simulated to design twenty cluster intelligent algorithms such as ant colony optimization algorithm, particle swarm optimization algorithm, artificial bee colony algorithm, pigeon group algorithm and bird group algorithm (Huo and Liu, 2017). Group intelligence algorithms can perform large-scale parallel computing. The range of applicability is broad and robust is strong. It is widely used in a variety of practical problems. Many domestic and foreign scholars have proposed many improved methods, but problems such as slow convergence and early maturity still need to be further studied (Im, et al., 2017). Previous researches have focused on group intelligence mechanisms and convergence, algorithm learning strategies, group intelligent hybrid algorithms and their application research. Combined with the homogenization and randomization methods, the population is initialized. For particle velocity updates, the perturbation is increased. For archive particles, the elimination mechanism is implemented (Maddi and Eslahchi, 2017). A multi-objective particle swarm optimization algorithm based on multi-strategy fusion is proposed. Through clustering, the entire population is divided into many subgroups. Two search mechanisms are introduced to interact with information within and between subgroups. An artificial bee colony algorithm based on information learning is proposed (Mao, et al., 2017). An artificial bee colony algorithm with adaptive extended memory is proposed to balance the global exploration and local development capabilities of the algorithm (Moeini, 2017). A hybrid algorithm of particle swarm and artificial bee colony is proposed. After all particles of the PSO are updated, the ABC update rule is performed on the individual extrema of the particle. From different perspectives, the iterative process of group intelligence algorithms, mutation, intersection and selection operators, and the balance between global exploration and local development capabilities are reviewed (Allan, 2017). Although a large number of research results have been obtained in the field of group intelligent optimization, the global exploration and local development ability of the equalization algorithm is still the key to solving the global optimization problem. While improving the convergence of the algorithm, falling into local

optimum should be avoided (Sejnoha, et al., 2017). Therefore, the group intelligence algorithm should be suitable for large-scale parallel optimization with good convergence performance and robustness. The PPI network is composed of a sub-network with a relatively independent topology or function. The proteins of the same functional module are densely connected to each other, and the proteins of different modules are sparsely connected to each other (Shunmugapriya and Kanmani, 2017). The PPI database is the primary resource for PPI network construction and biological pathway analysis. It is also an auxiliary means for the study of protein function. Therefore, in the analysis and prediction of mass PPI data, clustering analysis has become the main method and effective way for the prediction of PPI network function module (Zhang, et al., 2017). The effective clustering method can help to predict the function of unknown protein accurately.

The performance of the original functional stream clustering algorithm is improved by the search ability of the artificial bee colony (Bai, et al., 2017). A functional flow clustering algorithm based on artificial bee colony algorithm is proposed. The functional information flow simulation algorithm does not consider the effect of the distance. To solve this problem, an information flow clustering model based on bee colony optimization mechanism is proposed (Anusha and Sathiyamoorthy, 2017). A PPI network function module detection method based on multi-agent evolution mechanism is proposed. The agent in the algorithm completes the evolution by competing, intersecting and adaptively mutating the three operators to interact with the surrounding agents. A functional module detection algorithm based on multi-granularity description and ant colony optimization is proposed, which can quickly solve large-scale PPI networks. By coarsening the granularity, the PPI network is reduced in size and the running time is shortened. The pigeon population algorithm is used to optimize the PPI network clustering parameters to adapt to different sub-networks, thus successfully detecting the complex of dynamic PPI network functions (Chen, et al., 2017).

3. METHODOLOGY

3.1 Group Intelligence Algorithm Based on Adaptive Multi-Strategy Fusion

The group intelligent optimization algorithm is inspired by natural insects to propose a series of optimization methods for solving complex problems by simulating its group behavior. It has shown strong ability to search for optimization problems such as discontinuous, nonlinear, multi-constraint and non-convex, and has become a research hotspot of many researchers. The performance of the group intelligence algorithm is affected by the population initialization, problem size, and fitness function design. A number of strategies that facilitate global exploration or local development capabilities are organically integrated. Population diversity, escape stagnation/local extremum, accelerated convergence, and equilibrium of search capabilities are achieved. The specific ideas are as follows:

During the execution of the algorithm, when the individual distribution of the population is more dispersed, it is in the detection state. When the population is more concentrated, it is in a state of exploitation. The diversity of population distribution in the search space will result in a diversity of population fitness values. Moreover, the historical update of the individual fitness of the population helps to determine whether it is stuck in a stagnation/local extremum. Therefore, the use of uniform strategies and parameters for each dimension of the population will affect search efficiency. Therefore, adaptive multi-strategy fusion is studied by combining the distance distribution of populations on

smaller scales, search experience knowledge, fitness distribution and historical update. In the initial stage of the algorithm, the population model is selected by the initial population strategy to guide the population individuals to different regions of the search space for exploration to locate the optimal solution. The template vector is re-determined when the population's fitness history update is not ideal enough. When individual individuals fall into stagnation/local extremum, individual individuals are directed to other areas of the search space through strategies such as perturbation, mutation, chaotic search, or reinitialization. A new adaptive multi-strategy fusion method is proposed to detect the running state of the algorithm. The algorithm can dynamically update the key parameters of each strategy according to the evolution state of the population, so as to automatically invoke, convert and set the strategy. The initial assumption is to assign a uniform initial execution probability to each strategy. Then, in each iteration, the newly generated individuals are sorted according to their fitness, and each individual is assigned a weight in some way. For example, the weight can be set to:

$$\omega_i = \frac{\log(n-i+1)}{\log(1)+\log(2)+\dots+\log(n)} \quad (1)$$

Finally, these weights are added to the corresponding policy accumulator. After several iterations, the execution probability of each strategy is adjusted according to a certain calculation rule. In this way, each individual can select the appropriate strategy to complete the optimization according to the needs at different stages. The convergence of the algorithm is analyzed and proved by Markov chain or information entropy. Finally, the algorithm is applied to the optimization of complex optimization problems and benchmark functions for experimental verification and algorithm comparison to verify the effectiveness of the algorithm.

3.2 Heterogeneous Distributed Group Intelligent Hybrid Algorithm

Group intelligence algorithms are not susceptible to the scale and nonlinearity of the problem. It can provide new ideas and methods for the reduction of large-scale data. When solving complex problems, a single group intelligent algorithm is easy to fall into local optimum, and the generalization ability is weak and the precision is low. At the same time, considering that the single-group intelligent optimization algorithm cannot meet the processing requirements of massive data, cloud computing technology provides data analysis and processing technology for group intelligence algorithms. Therefore, according to the characteristics of group intelligence algorithm parallel computing, using the difference and complementarity of various group intelligence algorithms, the group intelligent hybrid algorithm is studied to improve the overall performance of solving complex problems. The proposed heterogeneous distributed group intelligent hybrid algorithm is designed as follows:

The population is first divided into N independent subpopulations. Each sub-population uses different population sizes, operation operators, and group intelligence algorithms with different search capabilities or different evolution methods of the same group intelligence algorithm. Each time a certain number of algebras are evolved, the subpopulations are partially transferred between the N subpopulations according to the designed migration rules, and so on, until the convergence condition is satisfied. In this way, each sub-group can enhance the diversity of the whole population by using group intelligence algorithms with different search capabilities or different evolution methods. Through the migration of some individuals between sub-populations, information sharing between individuals in the whole population is guaranteed, so that real-time information exchange between

algorithms can be realized to achieve synchronous evolution, and the algorithm has the ability of global parallel search and local serial search. However, the number and size of heterogeneous subpopulations, the group intelligence algorithm and evolution of heterogeneous subpopulations, the design of migration strategies between subpopulations, the choice of migration individuals and the timing of migration are key issues of this project. The structure of the heterogeneous distributed group intelligent hybrid algorithm is shown in Figure 1.

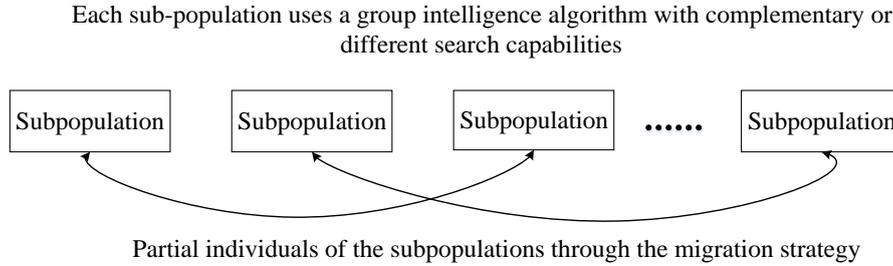


Figure 1. Co-evolutionary model of heterogeneous fractional group intelligent hybrid algorithm

The difference in evolutionary algorithms of each subpopulation is conducive to enhancing the diversity of the population. Using the diversity and complementarity of various group intelligent algorithms, the group intelligence algorithm with different global exploration and local development capabilities is selected as the evolution operation of the sub-population. Artificial bee colonies have strong global exploration capabilities, while particle swarm optimization and pigeon herd optimization algorithms have strong local development capabilities. Artificial bee colony and particle swarm or pigeon group and its improved algorithm are selected as subpopulation evolutionary algorithms, and the convergence of the algorithm is analyzed and proved by Markov chain or information entropy. According to the inherent parallelism of the algorithm, the Spark parallel computing framework is applied to build a parallel computing cluster to process large-scale and massive data.

The initial design of the migration strategy adopted is as follows: Individuals of each sub-population $P_i(i=1,2,\dots,N)$ use the group intelligence algorithm of each sub-group to optimize and participate in the competition. Its best individual is elected. In each γ generation, the migration strategy is used to exchange information between sub populations. The migration conditions are:

$$f(x_{best}^k) \leq f(x_{worst}^{k+m}) \quad , \quad m = 1,2,3,\dots \quad (2)$$

In the formula, k is the current subpopulation. When the migrated individual is superior to the replaced individual, the best individual in the sub-population is substituted for the worst individual or randomly selected individual of its neighbor sub-population. The influence of migration strategy parameters on the performance of the algorithm is studied to better understand the migration strategy and design a better cluster intelligent hybrid algorithm.

3.3 Dynamic PPI Data Clustering Integration Model Based on Group Intelligence Algorithm

PPI network data is massive and complex big data, and it has complex, high-dimensional and variable characteristics. Clustering or forecasting faces the challenge of a big data environment. Due to various data structure differences, noise and other situations, the clustering results are unstable and cannot

reflect the relationship between the data well. According to the clustering integration idea, the clustering integration is used to improve the accuracy of clustering results based on the improved clustering algorithm. The specific ideas are as follows:

The generated base clusters have different quality divisions, and the degree of difference between them varies. Therefore, it is the key to construct a clustering integration system by designing a basic clusterer with strong heterogeneity and generalization ability and selecting the appropriate number of base clusterers to participate in integration. An effective dissimilarity measure method also plays an extremely important role in building an integrated system. This project intends to use the output inconsistency measure to test the dissimilarity between clusterers. The classifier f_m and f_n dissimilarity measure $Diversity_{m,n}$ is expressed by the formula (3):

$$Diversity_{m,n} = \sum_{k=1}^M Dif(f_{mk}, f_{nk}) \quad (3)$$

In the formula, $Dif(f_{mk}, f_{nk})$ represents the difference between the output of the k th sample by the two classifiers. When the value is 0, it means that the output of the k th sample by the two classifiers is the same. When the value is 1, it means that the output of the k th sample by the two classifiers is different. Therefore, an output inconsistency matrix is obtained. $Select_m$ indicates the difference between the m th classification and all other classifiers, as shown in equation (4):

$$Select_m = \sum_{n=1}^N Diversity_{m,n} \quad (4)$$

The output inconsistency measure is a measure of dissimilarity that is tested based on the clusterer results. Therefore, on the basis of the above, the measurement of all the basic clusterers is considered and the influence of different measures on the clustering precision is studied in order to obtain the dissimilarity measure which is more suitable for clustering integration of dynamic PPI networks. The efficiency and generalization performance of the integration is further enhanced.

Integrated learning algorithms are also the key to affect the performance of clustering integration. Bagging uses a resampling method to improve the generalization ability of unstable base clusterers. In Boosting, the training set of each clusterer is affected by the previously generated clusterer, which may lead to over-fitting problems when dealing with actual problems. Therefore, the Baging method is used to generate the PPI network clustering model of the base clusterer through the Bag of Little Bootstraps technology, the parallel computing cluster strategy and the divide and conquer strategy. After generating the multi-cluster, if all of them are used to build an integrated clusterer, the prediction effect may be better than the single-based clusterer, but the prediction speed is significantly reduced. In order to achieve better prediction performance by using a small number of base clusterers, a selective integrated learning algorithm is used to remove the base clusterer that does not contribute much to clustering performance in the integrated clusterer. The base clusterer can be the same type or a different type of clusterer. The dynamic PPI data clustering integration model based on selective integration incremental learning is shown in Figure 2.

A heterogeneous group intelligent clusterer is used. In the multi-cluster integration process, each base clusterer is first given an initial weight using a selective integration method. The swarm intelligence algorithm is then used to optimize the weights assigned to each of the base clusterers, and the optimal base clusterer subset is selected. At the same time, combined with selective integration and

incremental learning, computational models and algorithms with incremental learning capabilities were studied. When a new sample is entered, only incremental learning is required, and no re-learning of all samples is required to achieve an update of the computational model. A PPI data clustering model based on integrated incremental learning with dynamic learning ability is proposed to enhance the dynamics of protein interaction and solve the problem of PPI massive data. In the selective integration, the reduction of the local-based clusterer set and the global integrated clusterer are mainly considered to improve the clustering performance and efficiency.

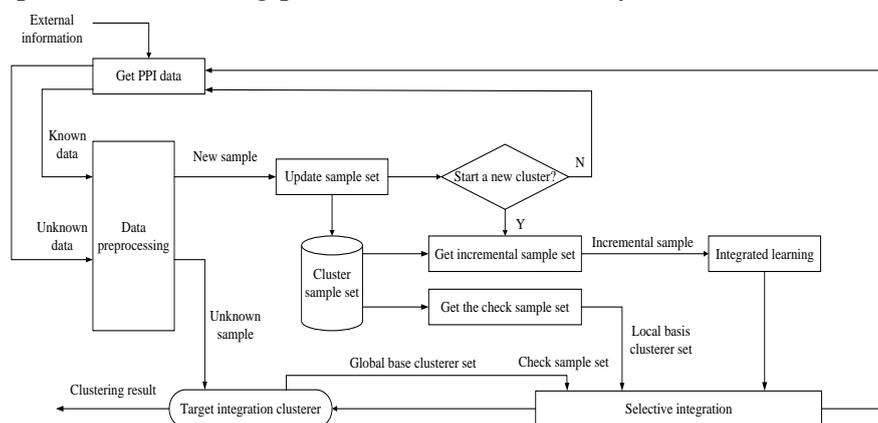


Figure 2. Dynamic PPI data clustering integration model based on selective integration incremental learning

The project relies on the unit computing environment and will provide a reliable equipment guarantee and material basis for the smooth implementation of the project. The Spark parallel computing framework has better speedup and scalability than the Map Reduce framework. It can handle large-scale or massive data better. Therefore, experimental tests and case studies are performed in the Spark parallel computing cluster environment. The experiment uses the python platform. The algorithm first verifies the complex optimization problem and the benchmark function and compares performance with other optimization methods. Then, on the large-scale datasets such as protein interaction DIP database, mammalian protein interaction MIPS database and gene co-expression data, the performance of PPI data clustering integration model and incremental learning model based on group intelligence algorithm was verified. Its performance is compared to the performance of other PPI data clustering methods. The above experimental platform and conditions can basically meet the requirements of the cross-over or parallel implementation of the experiment, which can ensure that the experimental project is implemented in stages according to the above research scheme.

4. RESULT ANALYSIS AND DISCUSSION

To test the performance of the cluster integration and incremental learning model under Spark on dynamic PPI data, the following experiment was carried out. The experiment was compared from two aspects of time performance and result evaluation parameters. In terms of time performance, the ACC-FMD algorithm was chosen as the comparison object. In terms of the result evaluation parameters, ACC-FMD, MCODE, MINE and Core four functional module detection algorithms are compared. The advantages and effectiveness of the ACC-MLF algorithm in the detection of large-scale PPI network functional modules are illustrated by comparison.

4.1 Comparison of Time Performance

The PPI database selects four data sets: Gavin, DIP core, MIPS and DIP full. The URL for the MIPS dataset is <http://mips.helmholtzmuenchen.de/proj/ppi/>. The parameter settings of the two algorithms on the four databases are introduced separately. The parameter settings for the ACC-MLF algorithm are as follows: On the Gavin and DIP core databases, $l=2$, $\epsilon=5$, $N=10$, $M=10$, $w=0.5$. On the MIPS database, $l=3$, $\epsilon=5$, $N=10$, $M=10$, $w=0.2$. On the DIP full database, $l=2$, $\epsilon=5$, $N=10$, $M=10$, $w=0.2$. The parameter settings for the ACC-FMD algorithm are as follows: on four databases, $kp = 0.9$, $kd = 0.2$, $\lambda = 3.0$, $N = 10$, $M = 10$. On the Gavin and DIP core databases, $w = 0.65$, $\delta = 0.05$. On the MIPS and DIP full databases, $w = 0.4$, $\delta = 0.02$. The ACC-MLF algorithm and the ACC-FMD algorithm are run on four databases respectively and compared in time to prove that the ACC-MLF algorithm is superior in the detection of large-scale PPI network function modules.

Table 1. Comparison of running time of two algorithms on four data sets

| Dataset | Networks size (number of nodes/number of edges) | Running time (seconds) | |
|----------|---|------------------------|-----------|
| | | ACC-MLF | ACC-FMD |
| Gavin | 1430/6531 | 110.929 | 2004.692 |
| DIP core | 2528/5728 | 125.923 | 887.175 |
| MIPS | 4545/12318 | 667.511 | 12768.525 |
| DIP full | 5032/22088 | 860.693 | 9132.770 |

Table 1 shows the running times of the ACC-MLF algorithm and the ACC-FMD algorithm on the four data sets. The first column of the table gives the name of the database. The second column gives the number of nodes and interactions in the database. The third column gives the time of the ACC-MLF algorithm. The fourth column gives the time of the ACC-FMD algorithm. As can be seen from the table, the running time of the ACC-MLF algorithm is greatly shortened compared to the running time of the ACC-FMD algorithm. Specifically, on the Gavin, DIP core, MIPS, and DIP full data sets, the run time of the ACC-FMD algorithm is 18.07 times, 7.05 times, 19.13 times, and 10.61 times of the ACC-MLF algorithm. Compared with the ACC-FMD algorithm, the ACC-MLF algorithm has a significant improvement in the time performance of functional module detection on a large-scale PPI network. It fully demonstrates the advantages of the ACC-MLF algorithm in the detection of large-scale PPI network functional modules.

4.2 Comparison of Detection Performance

The PPI database selects the MIPS data set and the DIP full data set. The parameter settings of the ACC-MLF, ACC-FMD, MCODE, MINE, and Core algorithms are introduced. On the MIPS database and the DIP full database, the parameters of the ACC-MLF algorithm and the ACC-FMD algorithm are the same as those of the time performance comparison section. The parameter settings of the MCODE and MINE algorithms are all set in the DIP full of the two algorithms in the third chapter of the experiment. The Core algorithm does not require any parameter settings. The results of the above various algorithms are compared on the values of coverage, sensitivity and accuracy. The effect of the ACC-MLF algorithm on clustering results on large-scale PPI networks was verified.

Figure 3 and Figure 4 show the comparison of clustering results of ACC-MLF, ACC-FMD, MCODE, MINE and Core algorithms on large-scale PPI datasets MIPS and DIP full, respectively. It is mainly explained by three specific evaluation indicators. From the first set of coverage data in the two figures,

it can be seen that the coverage of the ACC-MLF algorithm is 41.1% on the MIPS dataset, which is only 10.3% smaller than the ACC-FMD algorithm. In addition, its coverage is 30.2%, 16.2%, and 11.4% greater than the MCODE, MINE, and Core algorithms, respectively. On the DIP full dataset, the coverage of the ACC-MLF algorithm achieved a maximum of 73.7%, which was 16.6%, 59.1%, 36.2%, and 40.1% greater than the coverage of the ACC-FMD, MCODE, MINE, and Core algorithms, respectively. From the second set of sensitivity data in the two figures, it can be seen that the sensitivity of the ACC-MLF algorithm is 21.2% on the MIPS dataset, which is 4.9% and 1.8% smaller than the sensitivity of the ACC-FMD and MINE algorithms, respectively. It is 8.2% larger and 3.1% larger than the MCODE and Core algorithms. On the DIP full dataset, the sensitivity of the ACC-MLF algorithm is 44.4% maximum, which is 15%, 26.2%, 2.2%, and 12% greater than the sensitivity of the ACC-FMD, MCODE, MINE, and Core algorithms, respectively. From the third set of accuracy data in the two figures, it can be seen that compared with the other four algorithms, the ACC-MLF algorithm has no advantage in the accuracy value, and the effect is not as prominent as the other two evaluation index values.

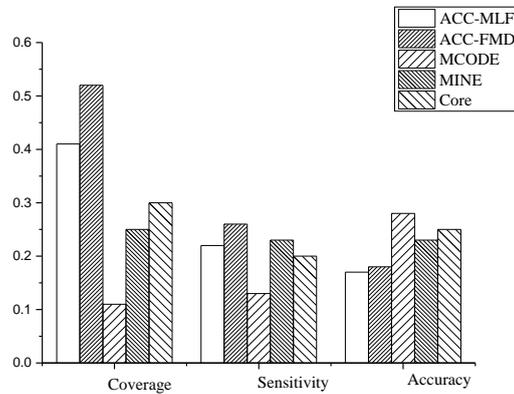


Figure 3. Comparison of evaluation values of clustering results of different algorithms on MIPS datasets

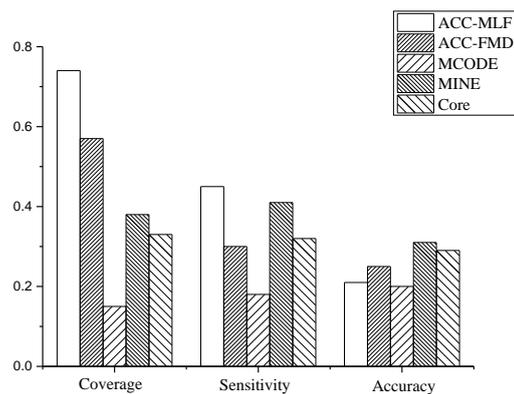


Figure 4. Comparison of evaluation results of clustering results of different algorithms on DIP full data set

By analyzing the data in Table 1, Figure 3, and Figure 4, the following conclusions can be drawn: In terms of runtime, the speed of the ACC-MLF algorithm is greatly improved. On the evaluation index value of clustering results, the clustering results of ACC-MLF algorithm have achieved good results

on some evaluation indicators. It proves the effectiveness of the algorithm in detecting PPI network function modules.

5. CONCLUSION

Adaptive multi-strategy fusion and heterogeneous distributed hybrid algorithms are studied. The performance of the group intelligence algorithm is improved. This has important application value and theoretical significance. In particular, the parallelism of group intelligence algorithms and the parallel computing power of the Spark framework are fully utilized. Large-scale data is processed efficiently. It has potential application value. The results have expanded the research and application of group intelligence and related technologies to a certain extent, so that they can achieve greater development and wide integration. The actual engineering optimization problem is effectively solved. The research results of PPI network clustering contribute to the development of disease treatment drugs. It provides the necessary technical means for disease diagnosis and classification. Dynamic PPI networks provide a more accurate understanding of biological pathways. It not only provides new ideas and decision-making references for the medical department and related researchers, but also provides important reference and technical support for new drug design and accurate medical treatment.

ACKNOWLEDGEMENTS

This work was supported by Wenzhou Vocational & Technical College scientific research project fund (no. WZY2018010).

REFERENCES

- [1] Allan, R. P. (2017). Decadal climate variability and the global energy balance. *Past Global Changes*, 25(1), 20-24. doi.org/10.22498/pages.25.1.20
- [2] Anusha, K., Sathiyamoorthy, E. (2017). MAM-ISSIDS: multi-agent model-based intelligent and self-sharing intrusion detection system for distributed network. *International Journal of Information and Computer Security*, 9(4), 361-386. doi.org/10.1504/ijics.2017.087567
- [3] Bai, W., Eke, I., Lee, K. Y. (2017). An improved artificial bee colony optimization algorithm based on orthogonal learning for optimal power flow problem. *Control Engineering Practice*, 61, 163-172. doi.org/10.1016/j.conengprac.2017.02.010
- [4] Chen, L., Xie, L., Yang, S. (2017). An algorithm to detect complexes in PPI network based on harmony search clustering optimization. *Journal of High Speed Networks*, 23(3), 215-223. doi.org/10.3233/jhs-170567
- [5] Huo, J., Liu, L. (2017). An improved multi-objective artificial bee colony optimization algorithm with regulation operators. *Information*, 8(1), 18. doi.org/10.3390/info8010018
- [6] Im, B. H., Jeon, D., Kim, C. H. (2017). Normalizers of intermediate congruence subgroups of the Hecke subgroups. *Open Mathematics*, 15(1), 787-799. doi.org/10.1515/math-2017-0066
- [7] Maddi, A. M. A., Eslahchi, C. (2017). Discovering overlapped protein complexes from weighted PPI networks by removing inter-module hubs. *Scientific Reports*, 7(1), 3247. doi.org/10.1038/s41598-017-03268-w
- [8] Mao, M., Duan, Q., Zhang, L. (2017). Artificial Bee Colony Algorithm Based on Adaptive Search Equation and Extended Memory. *Cybernetics and Systems*, 48(5), 459-482. doi.org/10.1080/01969722.2017.1319240

- [9] Moeini, R. (2017). Arc based ant colony optimization algorithm for solving sewer network design optimization problem. *Scientia Iranica. Transaction A, Civil Engineering*, 24(3), 953. doi.org/10.24200/sci.2017.4079
- [10] Sejnoha, M., Janda, T., Vorel, J. (2017). Combining Homogenization, Indentation and Bayesian Inference in Estimating the Microfibril Angle of Spruce. *Procedia engineering*, 190, 310-317. doi.org/10.1016/j.proeng.2017.05.343
- [11] Shunmugapriya, P., Kanmani, S. (2017). A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid). *Swarm and Evolutionary Computation*, 36, 27-36. doi.org/10.1016/j.swevo.2017.04.002
- [12] Zhang, Z., Lu, L., Zhang, Y. (2017). A combinatorial scoring function for protein–RNA docking. *Proteins: Structure, Function, and Bioinformatics*, 85(4), 741-752. doi.org/10.1002/prot.25253