

A Cadre Profiling Method Based on an Adaptive Two-Stage Mutation Genetic Attribute Reduction Algorithm

You-Ping Liu^{1*}, Jian-Gang Shen²

¹ Talent Assessment Department, China North Talent Research Institute, Building 26, Zone 3, Headquarters Base, Fengtai District, Beijing, China.

² Beijing Mafumatou Technology Co. Ltd., Building 5, Guofeng Meitang Building, Changping District, Beijing, China.

* Corresponding Author. Tel.: +86 13811548478; e-mail: bqrclyp@163.com

Manuscript submitted March 18, 2024; revised March 30, 2024; accepted April 18, 2024; published August 30, 2024

doi: 10.17706/jsw.19.3.86-97

Abstract: In the study of cadre profiling issue, the selection of label attributes poses the most significant challenge. In this paper, Rough Set is utilized to model the cadre profiling issue. Label selection issue, which stands for the difficulty of cadre profiling, is transformed into an attribute reduction issue within the Rough Set framework. The paper presents a Genetic Algorithm, which integrates adaptive crossover and mutation probability, best individual mutation and random mutation to address the attribute reduction issue. Comparative analysis demonstrates that the proposed algorithm exhibits good classification accuracy, attribute reduction rate and overall performance. Finally, the application effect of the proposed cadre profiling method is illustrated through an example of cadre selection and appointment.

Keywords: Rough set, attribute reduction, genetic algorithm, cadre profiling

1. Introduction

In recent years, cadre and talent profiling have been significant issues in human resources management. Numerous scholars have profiled cadres or talents using various methods, such as statistical analysis, recommendation systems, and artificial intelligence, solving practical problems such as recruitment, appointment, education, and retention. In the study of cadre profiling issues, the selection of label attributes poses the most significant challenge. Currently, most studies utilize expert prediction or empirical pattern recognition methods [1, 2], while others employ cluster analysis methods [3] and multi-modal subspace learning methods [4]. However, these approaches may suffer from drawbacks such as excessive reliance on expert experience, lack of objectivity, or excessive computational intensity for application to large datasets.

Rough Set is an artificial intelligence algorithm that analyzes data and infers implied knowledge therefrom. Attribute reduction is a crucial application area within Rough Set theory. It can identify essential attribute subsets from the set of all conditional attributes in the knowledge system, making the classification relative to the decision attributes derived from the attribute subset consistent with the classification derived from all conditional attributes, thereby reducing the number of conditional attributes.

In this study, we attempt to utilize Rough Set to model the cadre profiling issue, transform the label attribute selection issue into an attribute reduction issue, propose an attribute reduction algorithm based on an adaptive two-stage mutation genetic algorithm, and then illustrate the application of the algorithm

through the cadre selection and appointment scenario of a large group enterprise.

2. Basic Concepts and Modelling of Cadre Profiling

2.1. Basic Concept of Rough Set

The Rough Set theory characterizes the objective world using a knowledge system, defined as:

$$S = (U, A, V, f) \quad (1)$$

In Eq. (1), U is a non-empty object set called a domain. A refers to an attribute set of the research object in U , $A = C \cup D$, where C refers to a non-empty set composed of conditional attributes, D refers to a non-empty set comprising decision attributes, $C \cap D = \Phi$. V is the range, $V = \cup \forall a$, where $\forall a$ is the range of attribute a . f is the $U \times A \rightarrow V$ mapping, which assigns an informative value to each attribute of each research object.

In the knowledge system S , for each attribute subset $P \subseteq A$, the indiscernible relationship $IND(P)$ is defined as:

$$IND(P) = \left\{ (x, y) \mid \forall p \in P, f_{(x,p)} = f_{(y,p)} \right\} \quad (2)$$

The indiscernible relationship is also known as equivalent relationship.

For any object subset $X \subseteq U$ on domain U , the lower approximate set of X concerning attribute subset P is:

$$P(X) = \{x \in U, [x]_P \subseteq X\} \quad (3)$$

where $[x]_P$ is the equivalent class on the conditional attribute set P for object x .

In the knowledge system S , if P is a subset of conditional attributes and $P \subseteq C$, then the positive range of P for the decision attribute set D is:

$$POS_P(D) = \cup \{P(X) : X \in IND(D)\} \quad (4)$$

The dependency of decision attribute set D on the conditional attribute subset P , or the support of P on D is defined as:

$$\gamma_P(D) = \frac{card(POS_P(D))}{card(U)} \quad (5)$$

In Eq. (5), $card(POS_P(D))$ is the number of elements in the positive range of P for D and $card(U)$ is the number of elements in the domain, where $0 \leq \gamma_P(D) \leq 1$, the larger the value of $\gamma_P(D)$, the stronger the dependency of D on P .

Given a decision attribute set D , the reduction R for the conditional attribute set C can be defined as:

$$\gamma_R(D) = \gamma_C(D) \text{ AND } \gamma_{(R-a)}(D) \neq \gamma_R(D), (\forall a \in R) \quad (6)$$

For the given D , there is usually more than one reduction for the conditional attribute set C , and the intersection of all reductions is called the kernel of C and denoted $Core(C)$.

2.2. Modeling of Cadre Profiling Issue

The cadre profiling issue aims to identify label attributes that play a decisive role in decision-making in specific application scenarios from extensive cadre data and utilize these label attributes to characterize cadres. If Rough Set is employed to model the cadre profiling issue, then the issue of selecting label attributes can be addressed using attribute reduction algorithms.

Set the cadre profile knowledge system as $S = (U, A, V, f)$. The domain $U = \{x_1, x_2, \dots, x_n\}$ represents the collection of all cadres; any element x_i ($1 \leq i \leq n$) in U is a sample corresponding to a cadre object. $C = \{c_1, c_2, \dots, c_m\}$ represents the conditional attribute set, including gender, age, political affiliation, first education,

highest education, professional title, rank and so on. $D = \{d_1, d_2, \dots, d_k\}$ represents the decision attribute set and can be set according to the specific application scenario. Attribute reduction is carried out on the knowledge system S . The resulting reduced set R is the label attribute set of the cadre profile.

3. Attribute Reduction Algorithm Based on an Adaptive Two-Stage Mutation Genetic Algorithm

The traditional attribute reduction algorithm is mainly based on the FARNeMF algorithm [5–7], a forward greedy algorithm based on attribute importance proposed by Hu *et al.* However, these algorithms face two main challenges. One is that the computing process of the positive range cannot be carried out in parallel, leading to low algorithm performance. The other is that it is susceptible to local optimal solution. In recent years, some scholars have attempted to address the attribute reduction issues using heuristics such as Genetic Algorithms and Particle Swarm Algorithms [8–13], and made substantial progress in terms of classification accuracy, number of attribute reductions, and algorithm performance.

A two-stage mutation strategy—the combination of the mutation of best individuals and random mutation—is utilized in the Genetic Algorithm based on adaptive crossover probability and mutation probability, to address the attribute reduction issue. The algorithm is as follows:

3.1. Computing the Kernel

The kernel is the intersection of all reductions of a Rough Set, so the result of any attribute reduction algorithm must be a superset of the kernel. In this paper, at the start of the algorithm, the kernel $Core(C)$ of the Rough Set is calculated with the simplified difference function proposed by Zhou *et al.* [8]. In subsequent stages of population initialization, crossover, and mutation, each resulting new individual must contain all attributes in $Core(C)$, so as to narrow the search space and improve algorithm efficiency.

3.2. Coding Scheme

Binary encoding is utilized to encode individuals. Each individual of the Genetic Algorithm corresponds to a subset of the conditional attribute set C , i.e., a possible reduction. Each gene position of an individual corresponds to a conditional attribute one-to-one. If an attribute is included in the subset of conditional attributes, the value of the gene position corresponding to that attribute is 1. Otherwise, it is 0. For instance, if the conditional attribute set of a knowledge system is $C = \{c_1, c_2, \dots, c_{10}\}$, then the number of genes of each individual in the genetic algorithm population is 10. If a conditional attribute subset is $\{c_1, c_2, c_7, c_{10}\}$, the individual corresponding to this subset can be encoded 1100001001.

3.3. Fitness Function

Attribute reduction needs to consider two goals, that is, classification effect and reduction effect. The support degree $\gamma_B(D)$ of a conditional attribute subset B ($B \subseteq C$) for the decision attribute set D is a measure of classification effect, with larger values of $\gamma_B(D)$ indicating better classification effect. The reduction effect is measured by the number of attributes in B . The smaller the number of attributes, the more pronounced the reduction effect. Therefore, the fitness function of the attribute reduction issue can be set as:

$$Fit = a \times \gamma_B(D) + b \times \frac{card(C) - card(B)}{card(C)} \quad (7)$$

where $card(C)$ represents the number of attributes in the conditional attribute set C and $card(B)$ denotes the number of attributes in the subset B . a and b are the adjustment coefficients, with a larger value of a indicating more weight of the classification effect in the fitness function. This helps retain more knowledge information in the knowledge system, thereby aligning better with the Rough Set reduction criteria. Here,

we define $a = 0.8$, $b = 0.2$.

The $\gamma_B(D)$ part of the fitness function, involving computation of the positive range, is time-consuming. To improve the efficiency of algorithm, the fitness function is calculated solely for newly generated individuals in each generation during genetic algorithm execution. Moreover, for the newly developed individuals, multi-threading parallel computation of fitness values is employed for each individual.

3.4. Population Initialization

The population size has a dramatic impact on the performance and efficiency of Genetic Algorithms. When the size is too small, it results in poor gene diversity and challenges in achieving satisfactory results. Conversely, a too large size will increase operations and reduce efficiency. An and Liu [9] identified a more appropriate population size of 50 for solving attribute reduction issues using the Genetic Algorithm based on experiments and observations of 15 datasets with varying sample sizes and attribute numbers from the UCI database.

In this paper, we set the population size as 50 based on the Rough Set's kernel $Core(C)$ and randomly generated 50 individuals as the initial population. The steps for population initialization are as follows:

- (1) Set the value of the gene position corresponding to all attributes in $Core(C)$ to 1;
- (2) For attributes not in $Core(C)$, randomly set to 0 or 1;
- (3) Repeated steps (1) and (2) until all individuals reaching the population size are generated;
- (4) Calculate the fitness function for all individuals.

3.5. Selection

The selection process aims to choose individuals with higher fitness from the previous population to be inherited to the next generation, reflecting the idea of "survival of the fittest". The algorithm utilizes a standard roulette selection operator to provide individuals with high fitness a greater chance of advancing to the next generation. For the individual i , the probability $P_{(i)}$ of being selected is:

$$P_{(i)} = \frac{Fit_{(i)}}{\sum_{j=1}^n Fit_{(j)}} \quad (8)$$

In Eq. (8), $Fit_{(i)}$ denotes the fitness value of the individual i , while n represents the population size.

The selection process is:

- (1) Traverse all individuals in the population, calculate the selection probability P for each individual and the cumulative selection probability $Pacc$ of all individuals before the current one, and construct the layout of the roulette wheel. For the individual i :

$$Pacc_{(i)} = \sum_{j=1}^i P_{(j)} \quad (9)$$

- (2) Generate a random number r between 0 and 1 as a pointer on the roulette wheel;
- (3) If $Pacc_{(i)} \leq r < Pacc_{(i+1)}$, select the individual i ;
- (4) Repeat steps (2) to (3) until individuals reaching the population size were selected.

3.6. Crossover

The crossover operator generates new individuals in the genetic algorithm to realize the algorithm's global search ability. The standard Genetic Algorithm empirically utilizes a fixed crossover probability set during iterations and has certain blindness. From the overall evolution process of the population, the crossover probability should gradually decrease with the evolution process and eventually stabilize at a certain value to avoid disrupting the stability of the population in the late stage of the algorithm, making it hard for the algorithm to converge. Therefore, this algorithm employs an adaptive crossover probability:

$$P_c = \frac{\gamma}{1+e^{\alpha G}} + \beta \tag{10}$$

In Eq. (10), G represents the number of generation and α, β, γ is the constant coefficient. Considering that the crossover probability typically falls between 0.6 and 0.9, here we define $\alpha = 0.05, \beta = 0.6$ and $\gamma = 0.6$. The image of the crossover probability function is shown in Fig. 1.

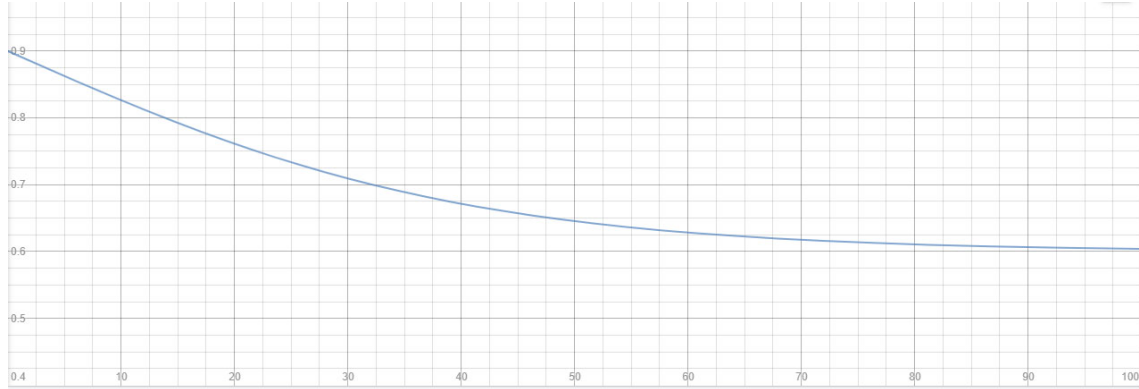


Fig. 1. Image of the crossover probability function.

Once the crossover probability P_c is determined, all individuals generated in the selection stage are traversed. Under the control of the crossover probability, individuals for crossover operation are identified. Subsequently, these selected individuals are grouped into pairs as parents, and the crossover operation is carried out.

Following the idea of “survival of the fittest”, individuals with higher fitness should contribute more genes to the offspring. Therefore, Eq. (11) is utilized to calculate the position of the crossover point:

$$Pos_{xover} = \frac{Fit_{(p1)} \times Card(C)}{Fit_{(p1)} + Fit_{(p2)}} \tag{11}$$

where $p1$ and $p2$ are the parents for crossover operation and $Fit_{(p1)}, Fit_{(p2)}$ represent the respective fitness values of the parents.

After Pos_{xover} is rounded to the nearest integer, the position of the crossover point is obtained. The swap of all genes of two individuals $p1$ and $p2$ after the crossover point generates two offspring individuals. After that, $p1$ and $p2$ are replaced. Let $p1 = 1100001001, p2 = 1101010101$, and Pos_{xover} is rounded to 5. The crossover between $p1$ and $p2$ is shown in Fig. 2.

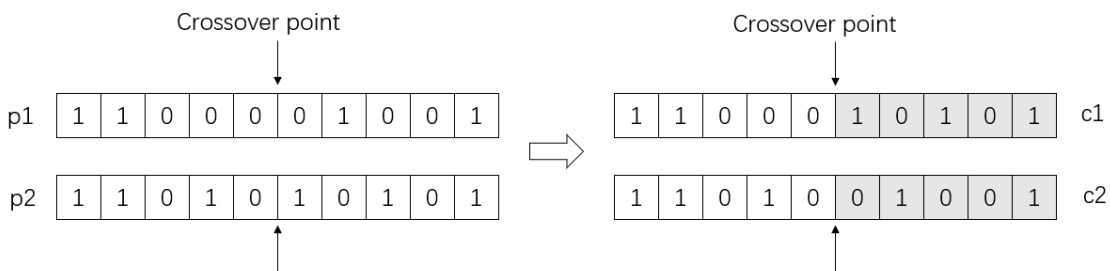


Fig. 2. Schematic diagram of cross operation.

3.7. Elite Retention Strategy

The elite retention strategy improves the performance and execution effect of the algorithm by preserving excellent individuals, so that the offspring can inherit excellent genetic information in the evolution process. After executing selection and crossover operations, the algorithm checks whether the

best individual from the previous generation is in the current population. If not, it replaces the worst individual in the current population with the best individual from the previous generation.

3.8. Two-Stage Mutation

Mutation provides diversity for the population's genes, enabling the genetic algorithms to "jump out" of local optimal solution, avoid premature convergence, and approach the global optimum. In the early stage of the algorithm, the population exhibits abundant diversity. In this case, the mutation probability should be lower to enhance the algorithm's operation speed. As evolution proceeds, individuals converge towards those with high fitness, resulting in a decrease in population diversity. In this case, the mutation probability should be larger, to maintain diversity. In this paper, Eq. (12) is utilized to calculate the mutation probability:

$$P_{mutate} = \frac{b}{1+e^{-aG}} + c \quad (12)$$

where G is the generation number, and a , b , c are constant coefficients. Considering that the mutation probability typically does not exceed 0.1, this algorithm sets $a = 0.05$, $b = 0.2$, and $c = -0.1$. The image of the mutation probability function is shown in Fig. 3.

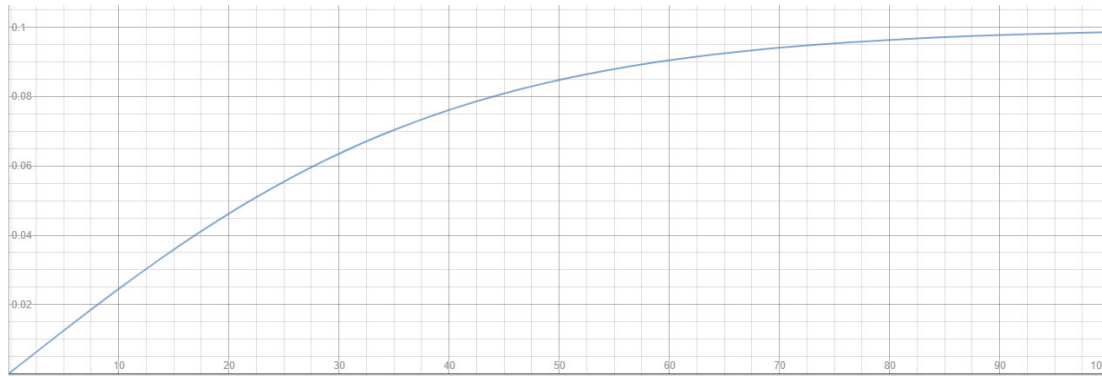


Fig. 3. Image of the mutation probability function.

This algorithm adopts a two-stage mutation strategy that combines mutation of the best individual and random mutation: under the control of mutation probability, the best individual is first mutated. If the mutation results in a higher individual fitness value, the worst individual in the current population is replaced with the new individual. If the mutation of the best individual fails, an individual is randomly selected to perform random mutation.

The mutation of the best individual adopts a single-point mutation strategy: traverse from the first gene position of the best individual and skip if the current gene position is in $Core(C)$. If not in $Core(C)$, the fitness value is calculated by inverting the value of this gene position. If it is greater than the fitness value before mutation, then the mutation is successful, and the worst individual in the population is replaced with the new individual generated in the mutation. Otherwise, the value of this gene position is restored, and the same operation is executed for the next gene position until the mutation is successful or all gene positions are traversed.

The process of random mutation is as follows: randomly select an individual from the population other than the best individual; randomly select several gene positions from the selected individual that are not in $Core(C)$. Invert the values of these gene positions, and if the fitness value of the resulting new individual is greater than the fitness value of the worst individual in the current population, the mutation is successful, and the worst individual is replaced with the new individual; otherwise, the selected individual is restored and another individual is re-selected for random mutation, until the mutation is successful or the number of

failed mutations reaches 5.

The mutation of the best individual is the fine-tuning of the best individual, which accelerates the “evolution” of the computational results towards a local optimal solution before reaching the local optimal solution. After reaching the local optimal solution, due to the small “magnitude” of mutation, it is difficult for the mutation of the best individual to produce better individuals. In this case, the introduction of random mutation can effectively guide the algorithm to “jump out” of the local optimal solution and “evolve” to the global optimal solution.

3.9. Algorithm Termination Criteria

The algorithm terminates when the fitness value of the best individual has not been optimized for 10 consecutive generations.

The algorithm flow is shown in Fig. 4.

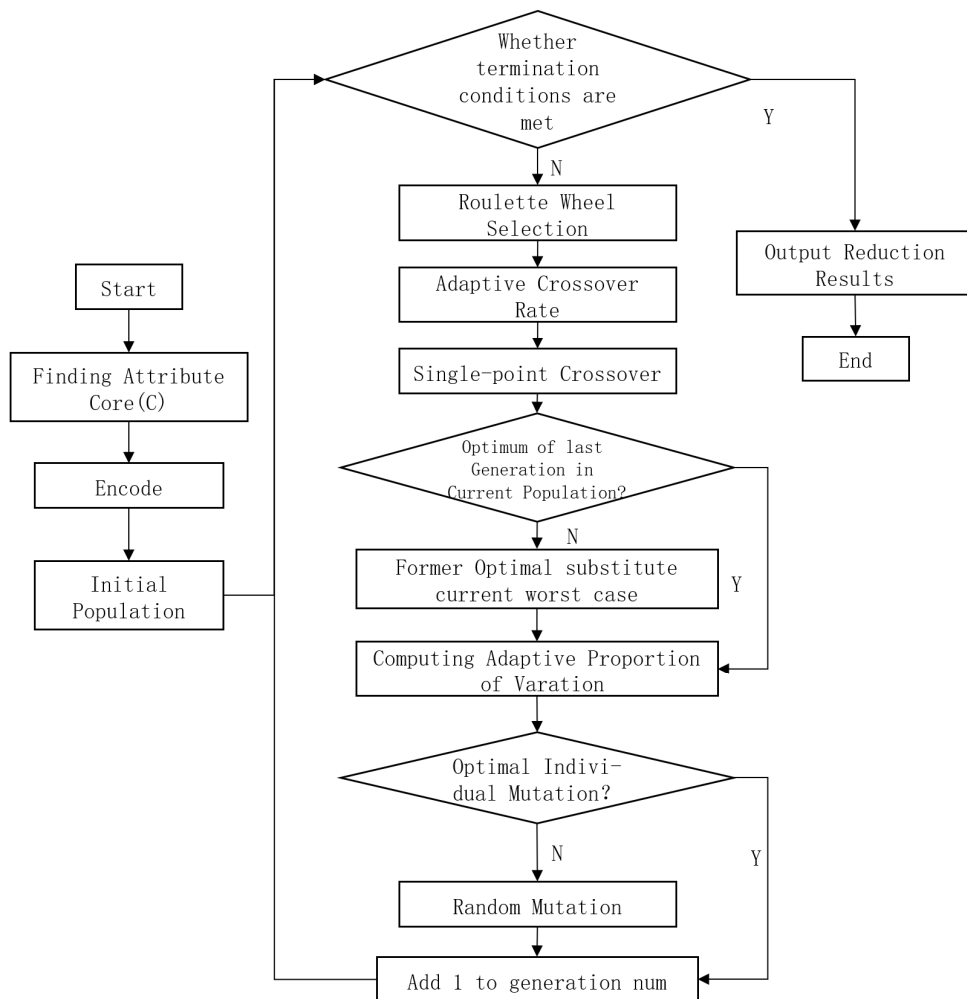


Fig. 4. Schematic diagram of algorithm flow.

4. Experiment and Application

4.1. UCI Dataset Comparison Experiment

From the UCI database, we selected three datasets with similar sample sizes and increasing numbers of attributes (from small to large). We compared them using the Partheno-genetic Algorithm in Attributes Reduction (PGAAR, proposed by Huang and Xiao [10]) and the Rough Set Attribute Reduction Algorithm based on Chaotic Discrete Particle Swarm Optimization (CBPSORS, proposed by Luan *et al.* [11]) in terms of

classification accuracy, attribute reduction rate, and run time. The basic information of three datasets is shown in Table 1.

Table 1. Information of the Experimental Datasets

Dataset	Number of Samples	Number of Conditional Attributes
Glass	214	9
Soybean (large)	307	35
Audiology	226	69

The experimental environment for the comparative analysis was a personal computer with an Intel Core i7 2.11 GHz CPU, 16 GB RAM, and Windows 10 operating system. In the experiment, we implemented three algorithms in Java and ran each algorithm 10 times to obtain the average for comparison. For PGAAR, the parameters described by Huang and Xiao [10] were used, while for CBPSORS, the parameters described by Luan *et al.* [11] were used.

4.4.1. Classification accuracy

Classification accuracy is defined in Eq. (13):

$$classification\ accuracy = \frac{number\ of\ samples\ correctly\ classified}{total\ number\ of\ samples} \times 100\% \tag{13}$$

The average classification accuracy of the three algorithms is shown in Fig. 5.

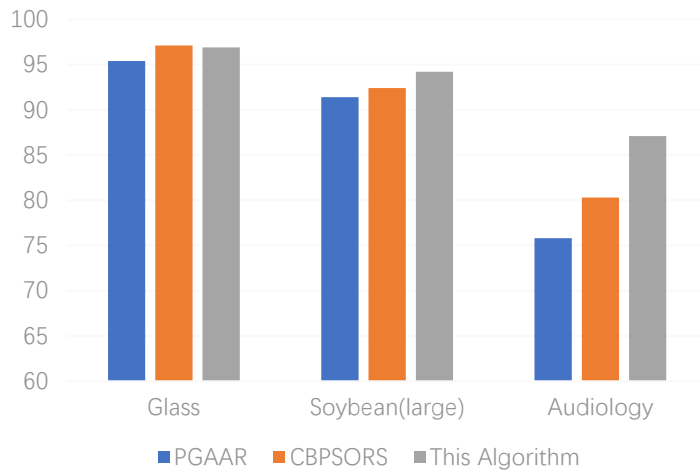


Fig. 5. Comparison of average classification accuracy among three algorithms.

From Fig. 5, when the number of conditional attributes is small, the three algorithms exhibit similar classification accuracy. When the number of conditional attributes is large, the proposed algorithm shows higher accuracy.

4.1.2. Attribute reduction rate

The attribute reduction rate is defined in Eq. (14):

$$attribute\ reduction\ rate = \frac{Card(C)-Card(B)}{Card(C)} \times 100\% \tag{14}$$

As the percentage of the reduced attributes in all conditional attributes. The average attribute reduction rates for the three algorithms are shown in Fig. 6.

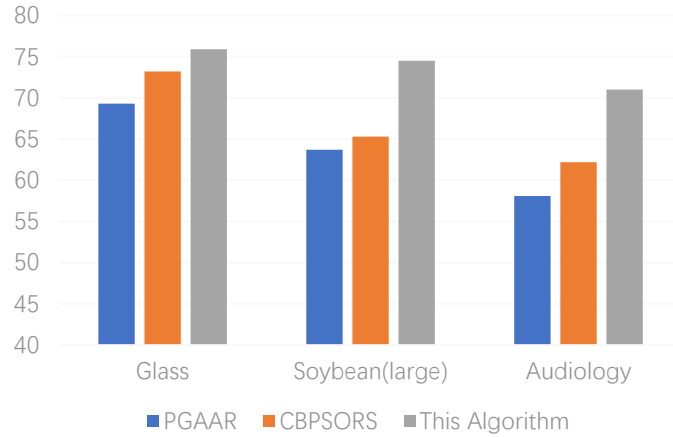


Fig. 6. Comparison of average attribute reduction rates among three algorithms.

As shown in Fig. 6, the proposed algorithm has a higher attribute reduction rate than those of the PGAAR and CBPSORS algorithms, especially when the number of conditional attributes is large.

4.1.3. Run time

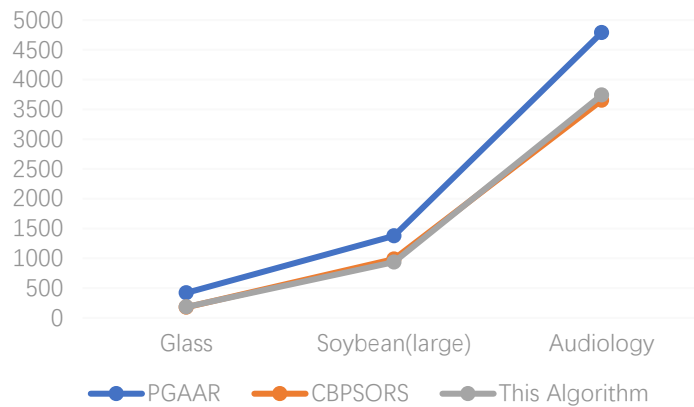


Fig. 7. Comparison of mean run time in ms among the three algorithms.

As shown in Fig. 7, CBPSORS and the proposed algorithm outperform PGAAR, and the proposed algorithm has comparable run times to CBPSORS.

To sum up, the proposed algorithm demonstrates comparable run time performance to CBPSORS. It has significantly better classification accuracy and attribute reduction rate than the other two algorithms, especially when the number of attributes is large.

4.2. Practical Application in Cadre Selection and Appointment Scenarios

Cadre selection and appointment are crucial application areas of cadre profiling. In a large-scale enterprise with nearly 200,000 employees, more than 30,000 management talents and more than 6,000 middle-level and above cadres, data from Internal Recruitment Platforms (IRP), Human Resource Management System (HRM), and Cadre Evaluation System (CES) are utilized as the raw data for cadre selection and appointment. The IRP contains 23,125 competition records and employment outcome of middle-level and above cadres from 2015 to 2023. At present, there are more than 4 million entries in the HRM system, from which information on cadre’s basic information, education and work experience, party and government positions, professional titles, awards and punishments, among others, are extracted. The CES contains more than 95,000 records of cadre evaluation results over the years. In this paper, we

extracted the evaluation results of leading cadres over the years. Since the Rough Set was not suitable for processing continuous data, we used the equal-frequency method to discretize continuous attributes such as age, height and weight. After preprocessing the original data using extraction, transformation, and loading tools, 66 conditional attributes were obtained, with the employment outcome (hired or not hired) as the decision attribute. The conditional attributes are listed in Table 2.

Table 2. List of Conditional Attributes

No.	Attribute	Origin
1	Gender	HRM
2	Age	HRM
3	Ethnic group	HRM
4	Native place	HRM
5	Place of birth	HRM
6	Registered residence	HRM
7	Place of file	HRM
8	Current residence	HRM
9	Length of service	HRM
10	Length of employment in the company	HRM
11	Political status	HRM
12	Length of party membership	HRM
13	Health status	HRM
14	Marital status	HRM
15	Professional title	HRM
16	Date of attainment of current professional title (months)	HRM
17	Current rank	HRM
18	Length of time in current rank (months)	HRM
19	Mode of appointment	HRM
20	Level of skill qualification	HRM
21	Personnel classification	HRM
22	Computer proficiency	HRM
23	English proficiency	HRM
24	First education	HRM
25	Field of specialization for the first education	HRM
26	Type of institution for the first education	HRM
27	Degree of the first education	HRM
28	Highest education	HRM
29	Field of specialization for the highest education	HRM
30	Type of institution for the highest education	HRM
31	Degree of the highest education	HRM
32	Length of time working at the grassroots unit (months)	HRM
33	Length of time working at the government agency (months)	HRM
34	Number of national-level honorary titles in the past 3 years	HRM
35	Number of provincial and ministerial-level honorary titles in the past 3 years	HRM
36	Number of group-level honorary titles in the past 3 years	HRM
37	Number of national-level awards in the past 3 years	HRM
38	Number of provincial and ministerial-level awards in the past 3 years	HRM
39	Number of group-level awards in the past 3 years	HRM
40	Number of unit rewards received in the past 3 years	HRM
41	Number of disciplinary actions received from the enterprises in the past 3 years	HRM

42	Number of disciplinary actions received within the party in the past 3 years	HRM
43	Number of publications in the past 3 years	HRM
44	Number of patents obtained in the past 3 years	HRM
45	Number of national projects undertaken in the past 3 years	HRM
46	Number of group-level projects undertaken in the past 3 years	HRM
47	Number of company-level projects undertaken in the past 3 years	HRM
48	Average score of comprehensive assessment in the past 3 years	CES
49	Average score of performance appraisal in the past 3 years	CES
50	Average score of party building assessment in the 3 years	CES
51	Average score of multidimensional evaluation in the past 3 years	CES
52	Average score of loyalty to the party in the 3 years	CES
53	Average score for boldness in innovation in the past 3 years	CES
54	Average score for ruling the enterprise effectively in the past 3 years	CES
55	Average score for promoting the enterprise successfully in the past 3 years	CES
56	Average score for integrity and honesty in the 3 years	CES
57	Height	IRP
58	Weight	IRP
59	Current income	IRP
60	Expected income	IRP
61	Whether there are relatives working within the group	IRP
62	Whether there are relatives overseas	IRP
63	Whether there is external funding from foreign countries	IRP
64	Whether there is a non-competition agreement	IRP
65	Whether there are labor disputes with other employers	IRP
66	Position applied for	IRP

Using historical data of 20,376 middle-level and above cadre selection and appointment records from 2015 to 2022 as the training set, we performed attribute reduction using the proposed algorithm and obtained 18 label attributes: age, gender, political status, first education, highest education, type of institution for the first education, professional title, current rank, length of time in current rank, personnel classification, length of time working at the grassroots unit, length of time working at the government agency, number of provincial and ministerial-level honorary titles in the past 3 years, number of group-level honorary titles in the past 3 years, average score of comprehensive assessment in the past 3 years and position applied for.

A Rough Set model was constructed to classify the test set using 2,749 competition records of middle-level and above cadres in 2023 as the test set, the above 18 label attributes as the conditional attribute set, and the employment outcome as the decision attribute. The classification result indicates that there were 475 individuals classified as “employed”. And in reality, 468 individuals were actually employed. There were 397 individuals that were both classified as “employed” and actually employed. The classification accuracy was 84.8%.

5. Conclusion

Cadre profiling is currently a vital research topic in enterprise management. In this paper, Rough Set theory is innovatively adopted to model the cadre profiling issue, and transform the label attribute selection issue into an attribute reduction issue. A two-stage mutation Genetic Algorithm based on adaptive crossover probability and adaptive mutation probability is proposed to address the attribute reduction issue. Through a comparative analysis with PGAAR and CBPSORS algorithms, the proposed algorithm can

achieve better classification accuracy and attribute reduction rate than PGAAR and CBPSORS algorithms within a run time comparable to CBPSORS. Finally, the application effect of the proposed cadre profiling method is illustrated through the cadre selection and appointment scenario of a large group enterprise.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

You-Ping Liu conducted the research. Jian-Gang Shen analyzed the data and wrote the paper. All authors had approved the final version.

References

- [1] Wang, X. X., Liu, J. S., and Xu, D. D. (2018). Operators' big data user profiling practices. *Telecommunication Science*, 34(5), 127–133.
- [2] Xue, L., & Song, W. (2020). Dynamic label-based relational extraction method. *Computational Machine Applications*, 40(6), 1601–1606.
- [3] Zhao, Y. G. (2021). Generation method of user profiling talent label system for big data mining. *Information Technology*, 12, 142–146.
- [4] Tian, F., Li, X., Liu, F., *et al.* (2020). Semantic label generation method based on multimodality subspace learning. *Journal of Shandong University: Engineering Edition*, 50(3), 31–37, 44.
- [5] Hu, Q. H., Yu, D. R., & Xie, Z. X. (2008). Numerical attribute reduction based on neighborhood granulation and rough approximation. *Journal of Software*, 19(3), 640–649.
- [6] Liu, Y., Huang, W. L., Jiong, Y. L., *et al.* (2014). Quick attribute reduct algorithm for neighborhood rough set model. *Information Sciences*, 271, 65–81.
- [7] Zhou, C. S., Xu, J. C., Qu, K. L., Shen, K. L., *et al.* (2022). A rapid attribute reduction method based on improving the importance of rough concentrated attributes in neighborhoods. *Journal of Northwestern University (Natural Science Edition)*, 52(5), 745–752.
- [8] Zhou, W., Feng, B. Q., & Liu, Y. (2007). Rough set efficient genetic reduction algorithm. *Journal of Xi'an Jiaotong University*, (4), 444–447.
- [9] An, L. P., & Liu, S. (2014). Two-stage genetic algorithm for attribute reduction. *Systems Engineering Theory and Practice*, (11), 2892–2899.
- [10] Huang, G. Q., & Xiao, S. (2018). Application of parthenogenetic algorithm in attribute reduction of rough sets. *Information Technology*, (5), 111–114.
- [11] Luan, Y. Y., Wang, X. H. & Xiao, J. M. (2021). Rough set attribute reduction algorithm based on chaotic discrete particle swarm. *Computer Simulation*, 38(7), 271–275.
- [12] Liu, Z. F., & Pan, S. (2021). Fuzzy rough set double reduction algorithm for particle swarm optimization. *Journal of Beijing University of Posts and Telecommunications*, 44(4), 49–55.
- [13] Fu, Z., & Pan, W. (2010). Radar fault diagnosis based on adaptive genetic reduction algorithm. *Control and Decision*, 4, 587–591.

Copyright © 2024 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).