Artificial Fish Swarm Algorithm for Mining High Utility Itemsets

Wei Song, Junya Li and Chaomin Huang
School of Information Science and Technology North China University of Technology, Beijing 100144, China
songwei@ncut.edu.cn

Introduction

Context: The discovery of high utility itemsets (HUIs) is an attractive topic in data mining. Because of its high computational cost, using heuristic methods is a promising approach to rapidly discovering sufficient HUIs. The artificial fish swarm algorithm is a heuristic method with many applications. Except the current position, artificial fish do not record additional previous information, as other related methods do. This is consistent with the HUI mining problem: that the results are not always distributed around a few extreme points.

Objective: The aim is to study HUI mining from the perspective of the artificial fish swarm algorithm and propose an HUI mining algorithm called HUIM-AF.

Methodology: We model the HUI mining problem with three behaviors of artificial fish: follow, swarm, and prey. We explain the HUIM-AF algorithm and compare it with two related algorithms on four publicly available datasets.

Results: The experimental results show that HUIM-AF can discover more HUIs than the existing algorithms, with comparable efficiency

Conclusion: These experiments show that the AFSA can leap over local optima effectively, so it is more suitable for the HUIM problem because there are multiple targets to optimize.

Research Contribution

A new HUIM algorithm based on the Artificial Fish Swarm Algorithm that is able to capture more HUIs than other two related algorithms

Problem Definition

In contrast to the support measure (which is used in FI mining), utility (used in HUIM) does not satisfy the downward closure property. Hence, the computational cost of HUIM is high. Furthermore, for application fields such as recommender systems, it is not necessary to use all HUIs.

The challenge is a 2-fold problem:To reduce the burden of HUIM, heuristic methods—such as genetic algorithm and particle swarm optimization (PSO) —have been used for HUIM, to discover acceptable itemsets within a reasonable time. For these algorithms, HUIs are discovered iteratively, and results of one iteration affect the HUIs discovered in the next iteration.

Thus, the resulting HUIs tend to be clustered around certain itemsets after many iterations, and the number of results is limited if no new individuals are generated randomly.

It was also verified in [9] that diversity is of great importance for generating a greater number of HUIs in a smaller number of iterations.

Postulations

In contrast to GA, PSO, and other heuristic methods that are used for HUIM, the artificial fish swarm algorithm (AFSA) only records the current position, but not other previous information, of each artificial fish (AF). This approach is essentially consistent with the problem of HUIM with a large number of diverse results. Therefore, in this paper, we use the AFSA to formulate an HUIM algorithm. The experiments show that the proposed algorithm can discover more HUIs than two other related algorithms

Assumptions

All 1-HTWUIs are sorted in a total order

The AFSA Method

The AFSA is inspired by the collective movement of fish and their typical social behaviors.

Preying Behavior. Preying is the behavior whereby a fish moves to a location with the highest concentration of food.

Swarming Behavior. In nature, a swarm of fish tends to assemble, so as to be protected from danger while avoiding overcrowded areas.

Following Behavior. When a fish finds a location with a higher concentration of food, other fish follow.

Modelling HUIM Using AFSA

We use a position vector (PV) to represent the position of an AF.
Letting HN be the number of 1-HTWUIs, a PV is represented by an HN-dimensional binary vector, in which each bit corresponds to one 1-HTWUI.
Assuming that all 1-HTWUIs are sorted in a total order, if the kth 1-HTWUI appears in a PV, then bit k of the PV is set to 1; otherwise, the bit is set to 0. It is proved that an item with a TWU value lower than the minimum utility threshold cannot appear in an HUI. Thus, only 1-HTWUIs are considered for representation in a PV.
Letting P be a PV, the jth ($1 \leq j \leq NH$) bit of P is randomly initialized using roulette wheel selection with the probability: we use the utility of the itemset directly as the object for optimization.
Letting X be the itemset corresponding to P, f (P) = u(X).

The HUIM-AF Algorithm

Input: Transaction database D, minimum utility value min_util, population size N, maximum number of iterations max_iter, maximum number of attempts try_number

Output: HUIs

```
Initialize N PVs using
SHUI = Empty Set;
iter = 1;
while iter less than or equal to max_iter do
    for i=1 to N do
is_follow = false;
is_swarm = false;
Pi = Follow(Pi);
if(!is_follow) then
    Swarm(Pi);
    end if
    if(!is_follow AND !is_swarm) then
        Prey(Pi);
    end if
    end for
        iter ++;
end while
Output HUIs.
```

Results

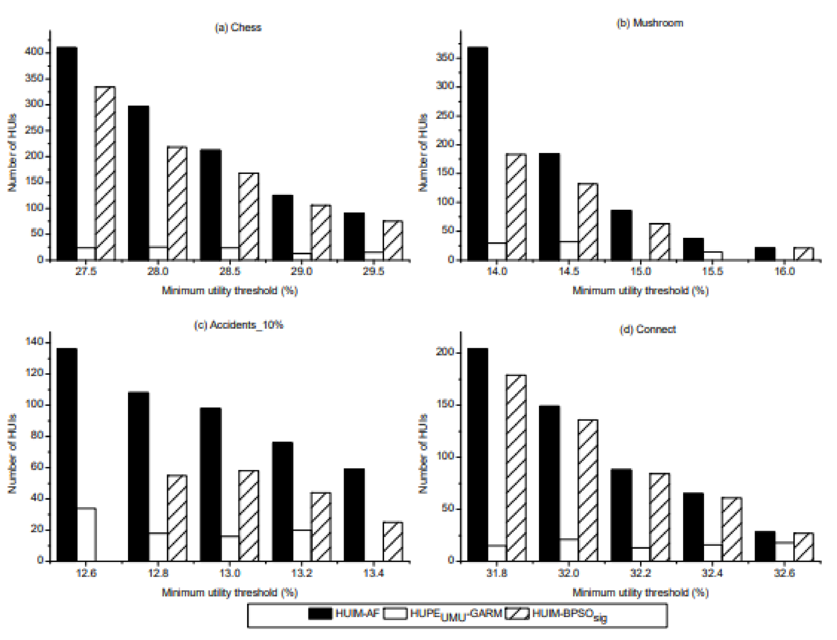

Fig. 1. Number of discovered HUIs for the four datasets

Conclusion

Our experimental results show that the AFSA can discover more HUIs than two other heuristic methods

Future Study

Our future work includes the design of effective pruning strategies to make this type of algorithm more efficient.

References

Fournier-Viger, P., Lin, J.C.-W., Gomariz, A., Gueniche, T., Soltani, A., Deng, Z., Lam, H. T.: The SPMF open-source data mining library version 2. In: Berendt, B., Bringmann, B., Fromont, É., Garriga, G., Miettinen, P., Tatti, N., Tresp, V. (eds.) ECML PKDD 2016. LNCS (LNAI), vol. 9853, pp. 36–40. Springer, Cham (2016). https://doi.org/10.1007/978- 3-319-46131-1_8

2. Fournier-Viger, P., Lin. J. C.-W., Vo, B., Nkambou, R., Tseng, V. S. (editors). High-Utility Pattern Mining: Theory, Algorithms and Applications, Springer, Cham (2019)

3. Kannimuthu, S., Premalatha, K.: Discovery of high utility itemsets using genetic algorithm with ranked mutation. Appl. Artif. Intell. 28(4), 337–359 (2014)

4. Li, X., Shao, Z., Qian, J.: An optimizing method based on autonomous animals: fish-swarm algorithm. Syst. Eng. Theor. Pract. 22 (11), 32–38 (2002) (in Chinese)

5. Lin, J. C.-W., Yang, L., Fournier-Viger, J., Wu, M.-T., Hong, T.-P., Wang, S.-L.L., Zhan, J.: Mining high-utility itemsets based on particle swarm optimization. Eng. Appl. Artif. Intel. 55, 320–330 (2016)

6. Liu, Y., Liao, W., Choudhary, A.: A two-phase algorithm for fast discovery of high utility itemsets. In: Ho, T.B., Cheung, D., Liu, H. (eds.) PAKDD 2005. LNCS (LNAI), vol. 3518, pp. 689–695. Springer, Heidelberg (2005). https://doi.org/10.1007/11430919_79

7. Song, W., Huang, C.: Discovering high utility itemsets based on the artificial bee colony algorithm. In: Phung, D., Tseng, V., Webb, G.I., Ho, B., Ganji, M., Rashidi, L. (eds.) PAKDD 2018. LNCS (LNAI), vol. 10939, pp. 3–14. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93040-4_1

8. Song, W., Huang, C.: Mining high average-utility itemsets based on particle swarm optimization. Data Sci Pattern Recognit 4(2), 19–32 (2020)

9. Song, W., Li, J.: Discovering high utility itemsets using set-based particle swarm optimization. In: Yang, X., Wang, CD., Islam, M. S., Zhang, Z. (eds.) ADMA 2020. LNCS (LNAI), vol. 12447, pp. 38–53. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-65390- 3_4