An Empirical Study of Observation-weighting Method for Mining Actionable Behavioral Rules

Peng Su¹², Yufeng Wang³*
¹Business School, Shandong University of Technology, Zibo 255012, China
²School of Mathematics and Computer Science, Dali University, Dali 671003, China
³Xingyuan primary school of Zhangdian District, Zibo 370303, China
Email: peng.su.casia@gmail.com, wangyufeng570@163.com

Abstract: Knowledge is considered actionable if users can take direct actions based on such knowledge to their advantage. Among a variety types of actionable knowledge, the actionable behavioral rule plays an important and unique role because it can directly and explicitly suggest actions to users to positively influence their behaviors of concerned entities. The problem of mining such rules is a search problem within a framework of support and expected utility. The previous mining approach assumes that each instance which supports a rule has the uniform contribution to the support. However, this assumption is usually violated in practice, and thus will hinder the performance of algorithms for mining such rules. To handle this problem, in this paper, we propose several observation-weighting models for support based on different functions. We further empirically investigate these models. Based on the results of our experimental study, we gain a thorough insight into the selection of various observation-weighting models for this domain.

1. Introduction
In recent years, data mining has become an important research area due to the exponential increase in the amount of data. With the popularization of computing and network technology in the world, the in-depth analysis and mining of data has become one of the most important means for people to create and accumulate domain knowledge and opinions.

Although the research on data mining technology has made great progress in recent years, only a small part of the published data mining algorithms has been successfully applied, and only a small part of the patterns mined in a large number meet the real interests and needs. The identified patterns lack the ability to directly recommend actions to users. This is mainly because there is a gap between academic objectives and real objectives, academic achievements and real expectations, technical evaluation system and real needs when applying data mining algorithms and technologies to problem solving and decision-making in the real world.

In order to bridge the gap between technology and reality, one of the most important real goals and expectations should be considered when applying data mining technology to the real world is to take action based on the mined patterns and ultimately benefit. For example, a bank would have a strong interest in such knowledge: "if the service level of male customers with high mortgage rate and low contribution is changed from low to high, 30% of such customers will become high contribution customers, and the bank will obtain a net income of about 10 million dollars." The ability
of knowledge to suggest actions benefiting users is called actionability [1 - 3], and knowledge with actionabilities is called actionable knowledge.

Previous research has looked into actionable knowledge discovery (AKD). For example, reference [4] proposes a formal view of actionable knowledge discovery from the perspective of system and decision-making, and formalizes and demonstrates the four types of general AKD framework.

AKD has been successfully applied in customer relationship management. References [5, 6] suggest a way to reclassify a customer from an unwelcome state to a popular state. The method reprocesses the decision tree to maximize the expected net income. In [7], some decision trees are processed to find the optimal actions. For these methods that need to integrate decision trees, the overlapping of attributes in multiple trees may significantly harm the knowledge acquisition in the next stage.

To solve this problem, a reordering method of decision tree is proposed in [8] to ensure that each tree is different at the level of parent and child nodes. In [9], a framework for postprocessing any additional tree model classifier to extract optimal actions is proposed. This framework is formally defined as an integer linear programming problem, and uses clear decision tree to mine actions. The main disadvantage of this method is that it is difficult to accurately estimate the effectiveness of the action. In order to solve this problem, the method based on fuzzy decision tree is used in [10] to find the optimal action.

In order to help a company design a direct marketing plan that can improve its revenue, a real-time method to generate suggestions and plans using role model is proposed in [11]. The role model is a typical case base that can generate user suggestions. For each new customer, nearest neighbor algorithm is used to find a plan with high net income and implementation probability to transform it into a popular role model. This method does not prepare the rules in advance, so the calculation cost of the action proposal is very high.

In order to improve the profitability of bank customers, action rules are proposed in [12]. Action rules are extracted from decision tables to describe the possible transition of some specific customers from the expected state (such as "non loyalty") to the expected state (such as "loyalty"). The proposed mining method of action rules is based on two kinds of classification rules which describe the expected state and the unexpected state respectively. First, attributes are divided into two categories: invariant attributes and mutable attributes. The former (the latter) contains attributes whose values cannot be changed or influenced by the enterprise. Then, the classification rules are extracted from the decision table in the way of variable attribute priority. Finally, a class of new rules called action rules is constructed from the classification rule base. A change in the value of a variable attribute indicates that an action has been taken. Action rules indicate how certain mutable attributes need to be changed to turn unwanted objects into popular objects.

There is a cost to changing the value of a variable attribute. In [13], the concept of feasibility and cost of action rules is proposed, and a method of constructing feasible action rules at the lowest cost based on search graph is proposed. However, there are two problems in the cost calculation model. On the one hand, only the cost of changing the variable attribute value is considered, but not the cost of corresponding change of the decision attribute value. On the other hand, the cost of a rule is defined as the average cost of reclassifying an object with the rule, and the average cost cannot reflect the total cost of the rule. To solve these two problems, a cost calculation model based on expected utility and two corresponding action rule mining algorithms are proposed in [14, 15].

The above methods of mining action rules are rule-based: Based on a pair of specific classification rules or a classification rule to produce an action rule. The main disadvantage of this mining method is that it may ignore some valuable rules of action. In order to solve this problem, in [16], a strategy of mining action rules directly from dataset samples under the framework of support confidence cost is proposed. In [17], a method of mining association action rules is proposed. In [18], a bottom-up strategy is proposed to mine action rules.
Previous studies have proposed a variety types of actionable knowledge. Among them, the actionable behavioral rule [19] plays an important and unique role because it can directly and explicitly suggest actions to users to positively influence their behaviors of concerned entities.

Consider an example in domain of security informatics. The actionable behavioral rule \((\{(e_1,0,1), (e_2,2,3)\},\{(b_1,2,2), (b_2,1,1/3), (b_3,2,0), 4/9\}), 0.7\) which is proposed to the Iran government against the Kurdistan Democratic Party suggests the following. “If the Iran government changes the degree of using lethal violence against the Kurdistan Democratic Party from level 0 (not using lethal violence) to 1 (using periodic lethal violence) and the degree of being in agreement with the Kurdistan Democratic Party from level 2 (negotiation) to 3 (minor concession), the degree of terrorist attacks aiming at domestic targets launched by the Kurdistan Democratic Party will remain unchanged with 2/9 confidence, or change from level 2 to 1 with 1/3 confidence or 0 with 4/9 confidence, and the Iran government will get the utility of 0.7.”

The aim of actionable behavioral rule mining is to identify all reliable and beneficial action proposals from a set of historical observations pertinent to a certain entity. The proposed definition of mining actionable behavioral rules explicitly states the problem as a search problem in a framework of support and expected utility [4]. The support of a rule is defined as the number of the historical observations which supports it. The threshold \(\text{minsup}\) is used to assure the significance of the rules.

Similar to other approaches of actionable rule mining and association rule mining, the previous work assumes that different observations supporting an actionable behavioral rule have the same degree of support for it. Therefore, the support of a rule is expressed as the total number of observations supporting it. However, because different observations have different temporal characteristics, their degrees of support for rules is not the same. Obviously, the smaller the distance of the period with regard to an observation from the current time is, the more support this observation provides to a rule (given that it supports the rule).

As stated above, the assumption of uniform contributions of observations will hinder the performance of the previous algorithms proposed to mine actionable behavioral rules.

To handle this problem, an observation-weighting model for support of actionable behavioral rule and corresponding mining algorithm was proposed [20]. More specifically, a linear function is constructed, which fit well the function associating the weight of an observation’s support with the distance of the period with regard to it from the forthcoming period. Furthermore, support of rule and support of effect-probability are redefined based on the proposed function.

Besides linear function, several functions can fit well the function associating the weight of an observation’s support with the distance of the period with regard to it from the forthcoming period. In this paper, we propose several observation-weighting models based on different functions. Furthermore, we empirically study the effects of these observation-weighting models on mining actionable behavioral rules. Based on the experimental results, we gain a thorough insight into the selection of various methods for this domain.

The rest of the paper is organized as follows. Section 2 describes our proposed observation-weighting models for support. Section 3 presents our empirical study which thoroughly investigates the performances of observation-weighting methods in this domain. Section 4 analyzes the observations and the results we obtain. Finally, Section 5 summarizes our contributions and the directions for future research.

2. Material and Methods

The precise quantitative relationship between the temporal characteristics of observations and their support for rules has not been studied and revealed. Therefore, our task is to construct a proper function which can fit well the function associating the weight of support of an observation \(o\) with the distance of the period with regard to \(o\) from the forthcoming period.

Let \(I = (O, o^*, A, D, \rho)\) be a behavioral information system. Assume that \(d_o \cdot D_{\text{max}}, D_{\text{min}}, w_o, \omega\) represent the distance of the period with regard to \(o\) from the forthcoming period,
MAX\{d_o|o \in O\}, MIN\{d_o|o \in O\}, the weight of o’s support, and \(w_o\) where \(d_o = D_{\text{max}}\), respectively. To fit well the function associating \(w_o\) with \(d_o\), the function \(F\) we will build should have three characteristics:

1. \(F(D_{\text{max}}) = \omega (0 < \omega \leq 1)\);
2. \(F(D_{\text{min}}) = 1\);
3. \(F\) is a monotonically decreasing function.

There are many functions meeting the above three limitations, such as power function, exponential function, etc. Unfortunately, it is unknown which function fits best the function associating \(s_o\) with \(d_o\). Therefore, we aim to use as many suitable functions as possible, and then choose the best one by an empirical study.

Let \(d\) denote \((D_{\text{max}} - d_o)/(D_{\text{max}} - D_{\text{min}})\), the functions we construct are described as follows:

**Power function:**
\[
F(d_o) = (1 - \omega)d^k + \omega \quad k > 0
\]

**Quadratic function:**
\[
F(d_o) = (1 - \omega)d^2 + \omega
\]

**Cubic function:**
\[
F(d_o) = (1 - \omega)d^3 + d^2 + d + \omega
\]

**Exponential function:**
\[
F(d_o) = (1 - \omega)(2^d - 1) + \omega
\]

**Log function:**
\[
F(d_o) = (1 - \omega)\log(d + 1)/\log 2 + \omega
\]

**Sine function:**
\[
F(d_o) = (1 - \omega)\sin(\frac{\pi}{2}d) + \omega
\]

**Asine function:**
\[
F(d_o) = (1 - \omega)\sin^{-1}(d) + \omega
\]

**Sec function:**
\[
F(d_o) = (1 - \omega)\sec\left(\frac{\pi}{3}d\right) - 1 + \omega
\]

**Asec function:**
\[
F(d_o) = (1 - \omega)\sec^{-1}(d + 1) + \omega
\]

**Tan function:**
\[
F(d_o) = (1 - \omega)\tan\left(\frac{\pi}{4}d\right) + \omega
\]

**Atan function:**
\[
F(d_o) = (1 - \omega)\tan^{-1}(d) + \omega
\]

Based on the definition of \(F\), we get the following observation-weighting model for support.

**Definition 1.** Let \(I = (O, o^*, A, D, \rho)\) be a behavioral information system. Observation-weighting support of action set \(S\) is defined as
\[ \text{wsup}(S) = \sum_{o \in O} (w_o \cdot \alpha). \]  

(12)

where \( \alpha = 1 \) if \( o \) supports \( S \), else \( \alpha = 0 \).

We call \( S \) a weight-frequent action set, or weight-frequent \( |S| \)-action set, with regard to a threshold \( \text{minwsup} \), if \( \text{wsup}(S) \geq \text{minwsup} \).

Definition 2. Let \( I = (O, o^*, A, D, \rho) \) be a behavioral information system. Observation-weighting support of an atomic actionable behavioral rule \( ar \) is defined as

\[ \text{wsup}(ar) = \sum_{o \in O} (w_o \cdot \beta). \]  

(13)

where \( \beta = 1 \) if \( o \) supports \( ar \), else \( \beta = 0 \).

Definition 3. Let \( I = (O, o^*, A, D, \rho) \) be a behavioral information system. Confidence of an atomic actionable behavioral rule \( ar \) is defined as

\[ \text{conf}(ar) = \frac{\text{wsup}(ar)}{\text{wsup}(S)}. \]  

(14)

Definition 4. Let \( I = (O, o^*, A, D, \rho) \) be a behavioral information system. Observation-weighting support of a candidate actionable behavioral rule \( cr = (S, C) \) is defined as

\[ \text{wsup}(cr) = \text{wsup}(S). \]  

(15)

3. Results

In this section, we empirically investigate the proposed observation-weighting methods for mining actionable behavioral rules using the MABR-3 algorithm proposed in [20].

3.1. Experimental Design

Based on 1789 samples of MAROB datasets, experiments are carried out to verify the effectiveness of the proposed method. The MAROB datasets covers a number of ethnic political organizations in the Middle East and North Africa. From 1980 to 2004, these datasets track and record multiple attribute values about these organizations every year. From the MAROB datasets, we extracted three sub datasets about Hezbollah in Lebanon, the Kurdistan Democratic Party in Iran and the Communist Party of Iraq for experiment. From the perspectives of corresponding governments, domain experts give the utility values of all possible actions and effects. These utility values are normalized to \([-1, 1]\).

The value range of parameter \( \omega \) is \((0,1]\). We take the discrete set \{0.1, 0.2, 0.3, 1\} to represent the value range.

3.2. Evaluation Criterion

Assuming that \( au \) is the actual utility when action set \( S \) does occur, and \( eu \) is the expected utility of actionable behavioral rule constructed by a method with \( S \) as the antecedent, the smaller the absolute difference between \( eu \) and \( au \) is, the more effective the method is. Therefore, we use the mean absolute error (MAE) as the criterion for evaluating the performance of different approaches. The MAE is given by

\[ \frac{1}{n} \sum_{i=1}^{n} |eu_i - au_i|. \]  

(16)

where \( eu_i \) is the expected utility an approach estimates and \( au_i \) is the actual utility.

3.3. Experimental Results

Table I shows the mean absolute errors for different support models with regard to different functions on the three sub MAROB datasets. When \( \text{minwsup} \) is greater than 3, the number of the rules with actual action sets as the antecedent the proposed approach produces will be too small.

---

1 The Minorities at Risk Organizational Behavior Datasets: http://www.cidcm.umd.edu/mar.
Therefore, \( \text{minwsup} \) is set to 3. When \( \omega = 1 \), each observations have the same weight of support for rules, that is to say, the proposed approach is equivalent to the previous one.

4. Discussion

It is worth noting that although the utility values of actions and effects are assigned by experts and inevitably subjective, in the experiment, the actual comparison is the absolute difference between the actual utility and expected utility of action sets. For the same action set, different methods will produce different effect estimation probability distribution. Generally speaking, the closer the distribution of effect estimation is to the reality, the closer the estimated utility is to the actual utility.

From Table I we can see that when \( \omega \neq 1 \), all the MAEs are less than 0.0917, the MAE of the previous method. This strongly suggests the validity and the superiority of our proposed approach.

| \( F \) | \( k \) | \( \omega \) |
|-------|
|       | 0.1  | 0.2  | 0.3  | 0.4  | 0.5  | 0.6  | 0.7  | 0.8  | 0.9  | 1    |
|       |      |      |      |      |      |      |      |      |      |      |
| 1/3   | 0.0658 | 0.0702 | 0.0721 | 0.0755 | 0.0779 | 0.0825 | 0.0830 | 0.0836 | 0.0841 | 0.0917 |
| 1/2   | 0.0631 | 0.0697 | 0.0717 | 0.0749 | 0.0774 | 0.0819 | 0.0825 | 0.0834 | 0.0840 | 0.0917 |
| 2/3   | 0.0590 | 0.0593 | 0.0624 | 0.0663 | 0.0766 | 0.0817 | 0.0821 | 0.0826 | 0.0831 | 0.0917 |
|       | 0.0590 | 0.0593 | 0.0625 | 0.0665 | 0.0769 | 0.0816 | 0.0822 | 0.0827 | 0.0831 | 0.0917 |
| 3/2   | 0.0636 | 0.0650 | 0.0714 | 0.0750 | 0.0775 | 0.0822 | 0.0830 | 0.0839 | 0.0852 | 0.0917 |
| 2     | 0.0613 | 0.0610 | 0.0650 | 0.0687 | 0.0792 | 0.0839 | 0.0845 | 0.0848 | 0.0842 | 0.0917 |
|       | 0.0533 | 0.0588 | 0.0624 | 0.0658 | 0.0723 | 0.0765 | 0.0803 | 0.0810 | 0.0827 | 0.0917 |
| 4     | 0.0484 | 0.0542 | 0.0597 | 0.0633 | 0.0692 | 0.0744 | 0.0788 | 0.0793 | 0.0816 | 0.0917 |
| 5     | 0.0496 | 0.0605 | 0.0647 | 0.0714 | 0.0690 | 0.0777 | 0.0812 | 0.0830 | 0.0849 | 0.0917 |
| 6     | 0.0464 | 0.0520 | 0.0579 | 0.0652 | 0.0680 | 0.0784 | 0.0782 | 0.0801 | 0.0822 | 0.0917 |
| 7     | 0.0389 | 0.0413 | 0.0480 | 0.0516 | 0.0591 | 0.0657 | 0.0788 | 0.0805 | 0.0811 | 0.0917 |
| 8     | 0.0370 | 0.0407 | 0.0497 | 0.0523 | 0.0579 | 0.0628 | 0.0775 | 0.0795 | 0.0808 | 0.0917 |
| 9     | 0.0430 | 0.0579 | 0.0630 | 0.0667 | 0.0702 | 0.0748 | 0.0782 | 0.0805 | 0.0823 | 0.0917 |
| 10    | 0.0431 | 0.0579 | 0.0633 | 0.0670 | 0.0702 | 0.0750 | 0.0780 | 0.0809 | 0.0824 | 0.0917 |
| 11    | 0.0434 | 0.0582 | 0.0637 | 0.0675 | 0.0708 | 0.0752 | 0.0781 | 0.0810 | 0.0826 | 0.0917 |
| 12    | 0.0516 | 0.0602 | 0.0670 | 0.0711 | 0.0739 | 0.0799 | 0.0810 | 0.0823 | 0.0837 | 0.0917 |
| Quadratic | 0.0639 | 0.0652 | 0.0715 | 0.0755 | 0.0780 | 0.0820 | 0.0838 | 0.0844 | 0.0852 | 0.0917 |
| Cubic  | 0.0632 | 0.0647 | 0.0722 | 0.0783 | 0.0777 | 0.0818 | 0.0840 | 0.0848 | 0.0850 | 0.0917 |
| Exponential | 0.0648 | 0.0695 | 0.0746 | 0.0764 | 0.0790 | 0.0802 | 0.0822 | 0.0837 | 0.0860 | 0.0917 |
| Log    | 0.0594 | 0.0615 | 0.0665 | 0.0764 | 0.0771 | 0.0818 | 0.0823 | 0.0828 | 0.0832 | 0.0917 |
| Sine   | 0.0362 | 0.0480 | 0.0575 | 0.0658 | 0.0723 | 0.0815 | 0.0819 | 0.0822 | 0.0827 | 0.0917 |
| Asine  | 0.0612 | 0.0633 | 0.0681 | 0.0738 | 0.0781 | 0.0810 | 0.0819 | 0.0826 | 0.0830 | 0.0917 |
| Sec    | 0.0541 | 0.0579 | 0.0611 | 0.0727 | 0.0733 | 0.0766 | 0.0789 | 0.0812 | 0.0827 | 0.0917 |
| Asec   | 0.0657 | 0.0716 | 0.0731 | 0.0788 | 0.0802 | 0.0827 | 0.0837 | 0.0840 | 0.0849 | 0.0917 |
| Tan    | 0.0649 | 0.0696 | 0.0711 | 0.0821 | 0.0810 | 0.0818 | 0.0833 | 0.0839 | 0.0847 | 0.0917 |
| Atan   | 0.0592 | 0.0615 | 0.0665 | 0.0763 | 0.0771 | 0.0818 | 0.0823 | 0.0827 | 0.0832 | 0.0917 |

When \( \omega = 0.1 \), all the models gain the best performance. In addition, as \( \omega \) goes up from 0.1 to 1, the MAE for nearly each model gets larger monotonically.

It is also shown that when \( (F, \omega) \) is set to (Sine, 0.1), the proposed method gains the best performance (MAE value is 0.0362). At this time, compared with the previous method, the proposed one achieves 60.5% performance improvement. This proves that the proposed method is significantly better than the previous one. In addition, when \( (F, \omega, k) \) is set to (power, 0.1,8), the suboptimal performance (MAE value is 0.0370) is obtained. At this time, compared with the previous method, the proposed one achieves 60.0% performance improvement.

It should be noted that in other domains, (sine, 0.1) is not necessarily the optimal value of \( (F, \omega) \). In other words, similar empirical studies are needed to obtain the optimal values of \( (F, \omega) \) for different domains.
5. Conclusions
Actionable behavioral rule mining has a broad application prospect in business intelligence, national security and other fields. In this paper, to resolve the problem of non-uniform contribution for different observations to support of actionable behavioral rule, we proposed several observation-weighting models for support. Based on the results of our experimental study, we gain a thorough insight into the selection of observation-weighting methods for this domain.

The models we construct are based on convex or concave functions in independent variable domain. In the future, we can use functions which are both convex in one part of independent variable domain and concave in the other part. In addition, more comprehensive experiments with many large datasets drawn from various domains can be conducted to validate the generalizability of our findings.

Acknowledgements
This work was supported in part by the National Natural Science Foundation of China under Grant No. 71462001.

References
[1] Liu,B., Hsu, W., Chen, S. (1997) Using general impressions to analyze discovered classification rules. In: the Third International Conference on Knowledge Discovery and Data Mining (KDD 97). pp. 31-36.
[2] Silberschatz, A., Tuzhilin, A. (1996) What makes patterns interesting in knowledge discovery systems. IEEE Transactions on Knowledge and Data Engineering, 8(6): 970-974.
[3] Kaur, H. (2005) Actionable rules: issues and new directions. Transactions on Engineering, Computing and Technology, pp. 61-64.
[4] Cao, L., Zhao, Y., Zhang, H., Luo, D., Zhang, C., Park, E.K. (2010) Flexible frameworks for actionable knowledge discovery. IEEE Transactions on Knowledge and Data Engineering, 22(9): 1299-1312.
[5] Ling, C., Chen, T., Yang, Q., Chen, J. (2002) Mining optimal actions for intelligent CRM. In: the Second IEEE International Conference on Data Mining (ICDM 02). pp. 767-770.
[6] Yang, Q., Yin, J., Ling, C., Chen, T. (2003) Postprocessing decision trees to extract actionable knowledge. In: the Third IEEE International Conference on Data Mining (ICDM 03). pp. 685-688.
[7] Alam, S., Alam, M. (2012) Actionable knowledge mining from improved post processing decision trees. In: the International Conference on Computing and Control Engineering (ICCCE 2012). pp. 1-8.
[8] Subraman, S., Wang, H., Balasubramaniam, S., Zhou, R., Ma, J., Zhang, Y., Whittaker, F., Zhao, Y., Rangarajan, S. (2016) Mining actionable knowledge using reordering based diversified actionable decision trees. In: the International Conference on Web Information Systems Engineering (WISE 16). pp. 553-560.
[9] Cui, Z., Chen, W., He, Y., Chen, Y. (2015) Optimal action extraction for random forests and boosted trees. In: the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 15). pp. 179-188.
[10] Kalanat, N., Shamsinejad, P., Saraei, M. (2015) A fuzzy method for discovering costeffective actions from data. Journal of Intelligent and Fuzzy Systems, 28: 757-765.
[11] Yang Q., Cheng, H. (2002) Mining case bases for action recommendation. In: the Second IEEE International Conference on Data Mining (ICDM 02). pp. 522-529.
[12] Ras Z., Tsay, L. (2003) Discovering extended action-rules (System DEAR). In: the International Intelligent Information Processing and Web Mining (IIPWM 03). pp. 293-300.
[13] Tzacheva A., Ras, Z. (2005) Action rule mining. International Journal of Intelligent Systems, 20(7): 719-736.
[14] Su, P., Li, D., Su, K. (2012) Mining valuable action rules: A social computing approach. Energy Education Science and Technology, Part B. Social and Educational Studies, 5(2): 995-1002.
[15] Su, P., Li, D., Su, K. (2012) An expected utility-based approach for mining action rules. In: the ACM SIGKDD Workshop on Intelligence and Security Informatics (ISI-KDD 12).
[16] He, Z., Xu, X., Deng, S., Ma, R. (2005) Mining action rules from scratch. Expert Systems with Applications, 29(3): 691-699.
[17] Ras, Z., Dardzinska, A., Tsay, L.S., Wasyluk, H. (2008) Association Action Rules. In: IEEE/ICDM Workshop on Mining Complex Data (MCD 08). pp. 283-290.
[18] Ras, Z., Dardzinska, A. (2008) Action rules discovery without pre-existing classification rules. In: RSCTC 2008 Conference. pp. 181-190.
[19] Su, P., Mao, W., Zeng, D., Zhao, H. (2012) Mining actionable behavioral rules,” Decision Support Systems, 54: 142–152.
[20] Su, P., Wang, L., Zeng, D., Liu, Y. (2015) An Observation-weighting method for mining actionable behavioral rules. In: the 2015 International Conference on Advanced Computational Intelligence.