DISCOVERY OF ACTIONABLE PATTERN IN BIGDATA USING INFORMATION GRANULES AND META ACTION WITH COST AND FEASIBILITY FOR EMOTION DETECTION

Angelina Tzacheva [1] , Sanchari Chatterjee[1] ,
Rajia Shareen Shaik[2] and Shiva Sai Praneeth Chakinala[2]

[1]Department of Computer Science and Information Technology,
Westcliff University, Irvine, CA 92612.
dr.tzacheva@gmail.com,
[1] Department of Computer Science, University of North Carolina at Charlotte,
Charlotte, NC, 28223.
schatt10@charlotte.edu,
[2] Software Engineering Division, Walmart Inc,
Bentonville, AR, 72713
rajia.shareen@gmail.com,
shivasaipraneeth.c@gmail.com

ABSTRACT

In the modern world of data, data mining focuses on techniques to extract surprising, engaging, and previously unknown patterns of knowledge from massive datasets. Extracting this data is beneficial in multiple domains. This paper explores Action Rules as a framework for extracting actionable insights from large-scale data in education and business. We introduce a Modified Hybrid Action Rule Mining approach with Information Granules and Meta-Actions. We assess the Cost and Feasibility of the discovered Action Rules. Our proposed method enhances scalability, efficiency, and interpretability through Big Data analytics. Experiments on student survey datasets and Net Promoter Score (NPS) business datasets demonstrate improved performance in transitioning emotions (e.g., Sadness to Joy, Detractor to Promoter). Our Results show that cost and feasibility of each Meta Action empower users to make informed, goal-oriented decisions.

KEYWORDS

Action Rules, Data Mining, Cost and Feasibility.

1. INTRODUCTION

In today's advanced technological world, the need for informed decisions and predictive systems is increasing across various industries, including education, business, medicine, retail, and finance. The relationship among the variables identifies the stable and variable features based on which action rules are generated. The action rules determine the transition of a state to the desired state. The methodology aims to convert data into actionable insights that drive decision-making. The Action Rule Mining literature consists of two major frameworks: the Rule-Based approach and the Object-based approach. In this work, we focus on the Modified Hybrid Action Rule mining method, which combines two Action Rule mining frameworks with the advantage of scalability with large datasets. Primarily, we emphasize Opinion Mining from Text to suggest Actionable Recommendations.

Rule-based learning [1] is a simple data mining method that identifies, learns, or develops 'rules' to store, operate, or apply. Association Rules and Decision Trees are fragments of rule-based methods that generate rules to associate patterns and classify data, respectively. In general, we constitute rules as given in Equation 1, where the antecedent is a conjunction of conditions, and the consequent is the resulting pattern in the provided data for the given conditions in the antecedent [30].

$$condition(s) \rightarrow result(s) \qquad (1)$$

An action rule is a process of knowledge extraction developed to advocate possible transitions for an individual to move from one state(negative) to another state (positive). For example, recommending the business to improve customer satisfaction [2] and sentiment analysis on Twitter [3]. Action rules follow the representation, similar to Equation 1, as given in Equation 2, where Ψ represents a conjunction of stable features, $(\alpha \rightarrow \beta)$ represents a conjunction of changes in values of flexible features and $(\theta \rightarrow \varphi)$ represents desired change in decision action which is beneficial to the user.

$$[(\Psi) \wedge (\alpha \rightarrow \beta)] \rightarrow (\theta \rightarrow \varphi) \qquad (2)$$

Action Rules recommending an Actionable pattern are prone to incur a definite cost to the user [4], [5]. Cost for actions in Action Rules include time, energy, money, or human resources. Actions being recommended can cause both positive(benefits) and adverse (losses) effects for users [6]. Thus, the Action Rules recommendations system should take on below cost low-cost to the users to make them plausible actions. The existing approaches [7–10] do not consider the cost-effectiveness of recommendations. In [4] [11], the concept of the cost of the Action Rules is introduced and refined. Searching for low-cost Action Rules from a large dataset can be very time-consuming and requires a distributed and scalable approach to extract them within a practicable timeframe.

In this paper, we worked on an extension to our previous work on distributed actionable pattern mining with Modified Hybrid Action Rule [12] mining approach that improves computational performance by combining the above two frameworks, thereby leveraging the scalability advantage for large datasets .We extract action rules from the business and survey datasets, which help to obtain better, desirable outcomes for the future, where a new Threshold Rho, which allows the user to choose the number of data partitions. We produced a table that talks about the cost and feasibility to achieve the desirable state. This yields Faster Scalable processing. We are applying the method to Student Survey Data; however, this method can be used for Improving Customer Satisfaction as well. We also aim to suggest ways to improve teaching methods and student learning, as well as how to change detractors into promoters in business. We implement and test our system in a Scalable Environment with Big Data using the Apache Spark platform.

2. RELATED WORK

Data Science plays a pivotal role in shaping the modern world [16]. The paper focuses on understanding its evolution, addressing challenges, and anticipating future trends, which are crucial for researchers, practitioners, and policymakers alike. Text mining can help identify patterns, trends, and relationships in text data that would be difficult or impossible to determine through manual analysis [29]. Natural Language Processing represents a cutting-edge technological paradigm with transformative impact for legal documentation. The paper [15] navigates the potential implications of employing NLP for legal documentation, emphasizing its role in improving access to justice, bridging linguistic gaps, and fostering inclusivity within the legal system.

In paper [28], the author combines probabilistic models, machine learning, and sentiment analysis to present a sophisticated method for recognizing and comprehending emotions in written communication. This approach predicts the likelihood of particular emotional states within textual data in addition to detecting subtle emotional nuances.

Emotion models are the foundation of the emotion detection process [25]. The paper highlights the different models of the Categorical Emotion Model, the Dimensional Emotion Model, and the Conditional Emotion Model. The paper [26] discusses keyword-based, rule-based, machine-learning-based, and deep-learning-based emotion detection approaches. The selection of approaches is based on the kind of emotions that are targeted to extract.

The paper [30] proposes a method that extracts action rules from data using a Genetic Algorithm to efficiently explore the ample search space, achieving significantly higher support and confidence. The traditional action rule mining DEAR methods rely on a two-stage process where classification rules are first mined and then transformed into action rules. These approaches are inherently limited by their dependence on classification models, which are primarily designed for prediction rather than for suggesting actionable changes. To overcome the drawbacks of irrelevance and ineffectiveness of action rules generated by such methods, GA2RM (Genetic Algorithm-Based Action Rule Mining) directly extracts action rules from raw datasets, eliminating the need for an intermediate classification step.

Ras and Tzacheva [4] introduced the concept of the cost and feasibility of Action Rules as an interesting measure. They proposed a graph-based method for extracting plausible and low-cost Action Rules. Ras and Tzacheva [4] proposed a heuristic search for new low-cost Action Rules, where objects supporting the latest set of rules also support the existing rule set, but the cost of reclassifying them is much lower under the new regulations. Later, Tzacheva and Tsay [11] proposed a tree-based method for extracting low-cost Action Rules. Some research, apart from Action Rules, has been done on extracting Actionable knowledge. Karim and Rahman [18] proposed another method to extract cost-effective actionable patterns for the customer attrition problem in the post-processing steps of Decision Tree and Naive Bayes classifiers. Su et.al [5] proposed a method to consider positive benefits that occur by following an Action Rule apart from all costs incurred from the same rule. Cui, et.al [19] proposed to extract optimal actionable plans during the post processes of the Additive Tree Model (ATM) classifier. These actionable patterns can change the given input to a desired one with a minimum cost. Hu et.al [20] proposed an integrated framework to gather the cost-minimal action sets to provide support for social project stakeholders to control risks involved in risk analysis and project planning phases.

Table 1. Example Decision System T

X	A	B	C	D
x_1	Y	N	N	D_1
x_2	Y	H	Y	D_2
x_3	Y	H	Y	D_1
x_4	N	N	N	D_2
x_5	N	H	N	D_1
x_6	N	N	Y	D_2
x_7	N	H	Y	D_2
x_8	N	H	N	D_1

In this work, we worked on our proposed method of Modified Hybrid Action Rule mining with an Additional Threshold Rho- for the Number of Partitions, which further improves the computational performance from our previous method that has only one threshold [12]. We produced a table that talks about the cost and feasibility to achieve the desirable state. This allows for Faster and more Scalable processing. We will apply our method to the Student Survey Data and NPS business data; however, this method can also be used to for healthcare data. We are focusing on our work to suggest ways to improve teaching and Student Learning methods, as well as customer satisfaction, such as transitioning detractors (Customers with Negative Emotions) to promoters (Customers with Positive Emotions) in business. We implement and test our system in a Scalable Environment with Big Data using the Apache Spark platform.

3. BACKGROUND

In this section, we provide a basic overview of decision systems, Action Rules, Spark and GraphX frameworks to help you understand our methodology.

3.1. Decision System

Consider a decision system given in Table 1. An information System can be represented as

$$T = (X, A, V) \tag{3}$$

where,

X is a nonempty, finite set of objects: $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$
A is a nonempty, finite set of attributes: A = A, B, C, D and,
V_i is the domain of attribute a, which represents a set of values for attribute $i|i \in A$. For example,

$$V_B = N, H.$$

An information system becomes a Decision system if

$$A = A = A_{St} \cup A_{Fl} \cup d \tag{4}$$

where D is a decision attribute.

The user chooses attribute d if they want to extract the desired action from $d_i : i \in V_d$. A_{St} is a set of Stable Attributes and A_{Fl} is a set of Flexible Attributes. For example, ZIPCODE is a Stable Attribute, and User Ratings can be a Flexible Attribute.

Let us assume from Table 1 that $C \in A_{St}$. A, B $\in A_{Fl}$ and D $\in$d and the decision maker desire Action Rules that trigger the decision attribute change from D_1 to D_2 throughout this paper for example.

3.2. Information System

Consider the information system given in Table 2. An information system can be represented as $Z = (X, M, V)$where, X is set of objects $\{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$ in the system; M is non-empty finite set of attributes $\{A,B,C,E,F,G,D\}$;V is the domain of attributes in M, for instance the domain of attribute B in the system Z is $\{B_1,B_2,B_3\}$.

Table 2. Information System Z

X	A	B	C	E	F	G	D
x_1	A_1	B_1	C_1	E_1	F_2	G_1	D_1
x_2	A_2	B_1	C_2	E_2	F_2	G_2	D_3
x_3	A_3	B_1	C_1	E_2	F_2	G_3	D_2
x_4	A_1	B_1	C_2	E_2	F_2	G_1	D_2
x_5	A_1	B_2	C_1	E_3	F_2	G_1	D_2
x_6	A_2	B_1	C_1	E_2	F_3	G_1	D_2
x_7	A_2	B_3	C_2	E_2	F_2	G_2	D_2
x_8	A_2	B_1	C_1	E_3	F_2	G_3	D_2

The information system in Table 2 becomes a Decision System if the attributes M are classified into flexible attributes M_{fl}, stable attributes M_{st} and decision attributes d, $M = (M_{st},M_{fl},\{d\})$. From table 2 $M_{st}= \{A, B, C\}$, $M_{fl}= \{E, F, G\}$, and $d = D$.

3.3. Action Rules

In this subsection, we give definitions of action terms, action rules, and the properties of action rules [21]

Let T = (X, A $\cup d$, V) be a decision system, where d is a decision attribute and V = $\cup V_i : i \in$A. Action terms can be given by the expression of $(m, m_1 \rightarrow m_2)$, where $m \in A$ and $m_1, m_2 \in V_m$. $m_1 = m_2$ if $m \in A_{St}$. In that case, we can simplify the expression as (m,m_1) or $(m = m_1)$. Whereas, m_1 6= m_2 if $m \in A_{Fl}$

Action Rules can take the form of $t_1 \cap t_2 \cap .... \cap t_n$, where t_i is an atomic action or action term and the Action Rule is a conjunction of action terms to achieve the desired action based on attribute D. Example Action Rule is given below: $(a,a_1 \rightarrow a_2).(b,b_1 \rightarrow b_2) \longrightarrow (D,D_1 \rightarrow D_2)$

3.3.1. Properties of Action Rules

Action Rules are considered interesting based on the metrics such as Support, Confidence, Coverage, and Utility. Higher these values, more interesting they are to the end user.

Consider an action rule R of form:

$$(Y_1 \rightarrow Y_2) \longrightarrow (Z_1 \rightarrow Z_2) \qquad (5)$$

where ,

Y is the condition part of R; is the decision part of R
Y_1 is a set of all left side action terms in the condition part of R
Y_2 is a set of all right-side action terms in the condition part of R
Z_1 is the decision attribute value on left side
Z_2 is the decision attribute value on right side

In [31], the support and confidence of an action rule R is given as

$$Support(\mathcal{R}) = min\{card(Y_1 \cap Z_1), card(Y_2 \cap Z_2)\}$$
$$Confidence(\mathcal{R}) = [\frac{card(Y_1 \cap Z_1)}{card(Y_1)}] \cdot [\frac{card(Y_2 \cap Z_2)}{card(Y_2)}]$$

Later, Tzacheva et al. [11] proposed new set of formula for the calculation of Support and Confidence of Action Rules. Their idea is to reduce the complexities in searching data several times for Support and Confidence of an Action Rule. The new formula is given

below.

$$Support(\mathcal{R}) = \{card(Y_2 \cap Z_2)$$
$$Confidence(\mathcal{R}) = [\frac{card(Y_2 \cap Z_2)}{card(Y_2)}]$$

Tzacheva et al. [11] also introduced the concept of utility for Action Rules. The utility of Action Rules takes the following form. In most of the cases, the Utility of Action Rules equals the Old Confidence of the same Action Rule.

$$Utility(\mathcal{R}) = [\frac{card(Y_1 \cap Z_1)}{card(Y_1)}]$$

Coverage of an Action Rule means how many decisions from values, from the entire decision system S, are being fully covered by all extracted Action Rules. In other words, using the extracted Action Rules, Coverage defines how many data records in the decision system can successfully transfer from Z_1 to Z_2

3.4. Cost of Action Rules

Generally, there is a cost associated with changing an attribute value from one class to another class, the more desirable one. The cost is a subjective measure, in the sense that domain knowledge from experts or users in the field is necessary to determine the costs associated with taking the actions. Costs can be moral, monetary, or a combination of both. For example, changing the marital status from 'married' to 'divorced' has a moral cost; whereas lowering the interest rate for a customer is a financial cost for the bank, in addition to any other monetary costs that may be incurred in the process. Feasibility is an objective measure, i.e. domain independent. According to the cost of actions associated with the classification part of the action rules, a business user may be unable or unwilling to proceed with them. The definition of cost was introduced by Tzacheva and Ras [4] as follows:

Assume that $S = (X, A, V)$ is an information system. Let $Y \subseteq X$, $b \in A$ is a *flexible* attribute in S and $v_1, v_2 \in V_b$ are its two values. By $\wp_S(b, v_1 \rightarrow v_2)$ we mean a number from $(0, \omega]$ which describes the average cost of changing the attribute value v_1 to v_2 for any of the qualifying objects in Y . These numbers are provided by experts. Object $x \in Y$ qualifies for the change from v_1 to v_2, if $b(x) = v_1$. If the above change is not feasible, then we write $\wp_S(b, v_1 \rightarrow v_2) = \omega$. Also, if $\wp_S(b, v_1 \rightarrow v_2) < \wp_S(b, v_3 \rightarrow v_4)$, then we say that the change of values from v_1 to v_2 is more feasible than the change from v_3 to v_4. Assume an action rule r of the form:

$$(b1 , v_1 \rightarrow w_1) \wedge (b2, v_2 \rightarrow w_2) \wedge ... \wedge (bp, v_p \rightarrow w_p) \Rightarrow (d, k_1 \rightarrow k_2)$$

Table 3. Meta-actions Influence Matrix for S

	a	**b**	**d**
M_1, M_2, M_3		$(b_1 \rightarrow b_2)$	$(d_1 \rightarrow d_2)$
M_1, M_3, M_4	(a_2)	$(b_2 \rightarrow b_3)$	
M_5	(a_1)	$(b_2 \rightarrow b_1)$	$(d_2 \rightarrow d_1)$
$M2, M4$		$(b_2 \rightarrow b_3)$	$(d_1 \rightarrow d_2)$
$M1, M5, M6$		$(b_1 \rightarrow b_3)$	$(d_1 \rightarrow d_2)$

If the sum of the costs of the terms on the left-hand side of the action rule is smaller than the cost on the right-hand side, then the rule r is feasible.

3.5. Meta Action

An action rule can be seen as a set of atomic actions that need to be executed to achieve the expected result. Meta-actions on the other hand, are the actual solutions that should be executed to trigger the corresponding atomic actions. Table 3. above shows an example of an influence matrix that describes the relationships between the meta-actions and atomic actions influenced by them.

3.6. Spark

Spark [13] is a framework that is quite similar to MapReduce [12] for processing large quantities of data in a parallel fashion. Spark introduces a distributed memory abstraction strategy called Resilient Distributed Datasets (RDD) that can perform in-memory computations on nodes distributed in a cluster. The results of each operation are then stored in memory, which can be accessed for future processes and analyses, creating another RDD in turn. Thus, Spark reduces the number of disk accesses for storing intermediate outputs, similar to Hadoop MapReduce. Spark functions in two stages: 1. Transformation, 2. Action. During the Transformation stage, computations are performed on data splits, and the results are stored in the worker nodes memory as RDD. While the Action stage on an RDD collects results from all the workers and sends them to the driver node, or saves the results to a storage unit. With RDDs, Spark helps machine learning algorithms skip innumerable disk accesses during iterations.

4. DATASET DESCRIPTION

To test our methods, we use two datasets: Student Survey Data [14] and the Net Promoter Score dataset [17]. Student survey data aim to evaluate students' emotions and overall satisfaction with course teaching methods and group work experiences. The survey is designed to gather meaningful insights into students' feelings towards Active Learning methods and other factors that can aid students in their learning process. The data is collected in the courses that implement the Active Learning methods and teaching style. This survey dataset contains 50 attributes. The original data includes 549 instances and 59 attributes. Data is collected in classes employing Active Learning methods to assess students' opinions about their learning experience in the years 2019 and 2020. The data size on disk is 59 Kilobytes. For scalability purposes to test the performance of our proposed method with Big Data, we replicate the original Student Survey Data 100 times. The replicated dataset has a total of 54900 instances. Size on disk is 5.815 Megabytes. We also used a sample of the Net Promoter Score dataset [17] for our experiments. The NPS (Net Promoter Score) dataset is collected from customer feedback data related to heavy equipment repair. The entire dataset comprises of 38 companies, located at multiple sites across the United States and several parts of Canada. The decision attribute in the dataset is PromoterStatus which labels each customer as either promoter, passive or detractor. The decision problem here is to improve customer satisfaction / loyalty as measured by the Net Promoter Score. The goal of applying action rules to solve the problem is to find minimal sets of actions so that to "reclassify" customers from "Detractor" to "Promoter" and improve NPS. For our experiments, we used a survey completed by customers of two companies in 2015. We have used 17-California and 30-35 datasets for our method. Each NPS dataset consists of approximately 1,500 unique surveys from multiple customers, each containing around 25 distinct questions. The original data for 17-California contains 547 instances and 23 attributes, and the dataset for company 30-35 includes 3335 cases and 23 attributes.

5. METHODOLOGY

In our paper, we produced a table containing the cost and feasibility of each Action Rules along with the Meta Actions.

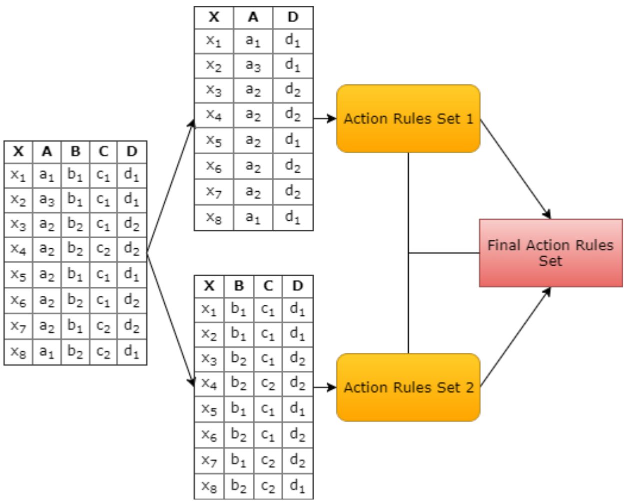

Figure 1. Example Vertical Data Distribution for Table 1

5.1. Modified Hybrid Action Rule Mining with Partition Threshold Rho

We propose Modified Hybrid Action Rule Mining Algorithm with a Partition Threshold Rho which provides scalability for big data. It presents a significant improvement over the previous method, Hybrid Action Rule Mining, which has a several disadvantage. If the Size of the Intermediate Table becomes huge, it affects the performance and the scalability of this method. Our proposed new method addresses this problem, as the Threshold ρ enables the user to control the size of the table, thereby increases the computational speed.

Our proposed method, Modified Hybrid Action Rule Mining with Partition Threshold Rho, is presented in Figure 3 and the proposed methodology is depicted in Figure 2.

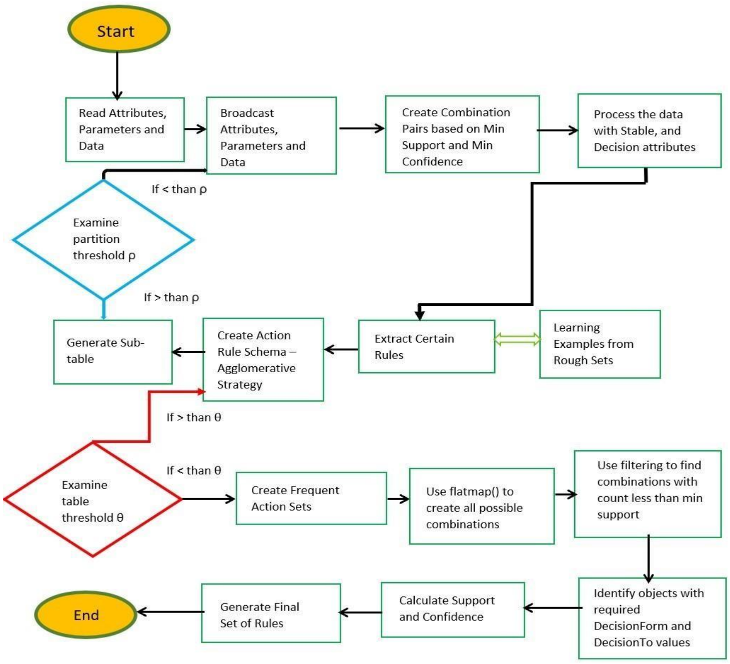

Figure 2. Hybrid Action Rule Mining Algorithm (New Threshold) – Flowchart

```
Algorithm(certainRules, decisionFrom, decisionTo, support, confidence)
        (where certainRules are provided by algorithm LERS)
        for each rule r in certainRules
            if consequent (r) equals desisionTo
                    Form ActionRuleSchema (r)
                    ARS <- ActionRuleSchema (r)
            end if
        end for
        for each schema in ARS
                Identify objects satisfying schema
                Form partition
                While partition size > Rho ρ
                Form subtable
                While subtable size > Theta θ
                    Divide subtable until subtable < Theta θ
                Generate frequent action sets using Apriori
                Combine frequent action set to form Action Rules
                (such that the frequent action sets satisfy the
                        decisionFrom -> decisonTo)
                Output <- Action Rules
        end for
```

Figure 3. Hybrid Action Rule Mining with Threshold Algorithm.

5.2. Vertical Data Distribution Method with Meta Action

Meta Actions, are a tabular format to trigger action rules discovered from user data. Meta Actions are the actions that need to be executed to trigger corresponding [22] actions, which can be one or more sets to invoke action rules in our method. A set of Meta Actions triggers the generation of action rules.

In our paper, we present an approach for partitioning the given data using information granules. We present a new algorithm to generate Meta Action as the intermediate state before extracting of all Action Rules, based on the algorithms proposed in [23] and [9]. To overcome the expense and computational complexity, the authors in [21] propose a vertical data split method for parallel processing, which enables along with faster computation. In this method, the data is split vertically order into two or more partitions, with each partition containing only a small subset of the larger attributes. Figure 1 illustrates the example of Data partitioning using Vertical Data Distribution in the Distributed Action rules extraction algorithm, the first section of the methodology.

Figure 1 presents an example of vertical data partitioning using the sample Decision system in Table 1. The actionable knowledge extraction algorithm runs separately on each data partition, performs transformations such as map() and flatmap() functions and combines results using join() and groupBy() operations. We later combine action rules from different partitions to get the final set of action rules.

We test the speed of our new method using two different datasets: one based on the NPS dataset and the other on the Student Survey dataset, and compare it to our previous distributed Action Rule extraction algorithms. A brief description of our vertical data distribution process with Meta Actions is provided in Figure 4. We validate the new data distribution method by comparing the number of Action Rules generated by our method with the rule coverage of Action Rules from the system, as well as classical Association Action Rules [9] on a single machine and SARGS [15] systems.

6. EXPERIMENTS AND RESULTS

In this work, we use student survey data that focuses on student emotions and NPS (Net Promoter Score) data [21]. for our experiments, which aim to evaluate Promoter Status.

We generated tables, in addition to our previous work (CITE) to display the cost and feasibility for each Action Rules for NPS and Student Survey data.

6.1. Experiment 1 - Modified Hybrid Action Rule Mining with Partition Threshold Rho Implementation in Spark AWS Cluster

We perform this experiment on the Student Survey Data using our proposed Modified Hybrid Action Rule Mining Method, which utilizes Amazon Web Services (AWS) cluster with two nodes, 4 vCore and 16GB memory along with EBS storage of 64GB. We have created an extension to our previous work by generating Cost and feasibility for the Action Rules as shown in Table 4.

Table 4. Action Rules of Student Survey datasets: Sadness to Joy ::: - Student Survey Data - Hybrid Method with Threshold

Experiments	**Meta-Action**	**Cost (0-1)**	**Feasibility (0-1)**
1.AR1SadnesstoJoy:(TeamSenseofBelonging,2BelowAverageSenseofBelongingtotheTeam → 3AverageSenseofBelongingtotheTeam) ∧ (NumberofTeamMembers,5to7 →10orMore) =⇒ (StudentEmotion,Sadness →Joy)[Support : 20.0,Confidence : 59.0%]	Increase the sense of belonging for each student in group activities	0.80	0.7
2. AR2SadnesstoJoy:(NumberofTeamMembers,5to7 → 8to10) ∧ (TeamWorkHelpedDiversity,2Occasionally → 3Often) ∧ (GroupAssignmentBenefit,None → AllofThem) =⇒ (StudentEmotion,Sadness→Joy)[Support:20.0,Confidence : 100%]	Increase diversity in group assignments	0.70	0.8
3. AR3SadnesstoJoy:(NumberofTeamMembers,5to7 → 8to10) ∧ (GroupAssignmentBenefit,None →SharedKnowledge) =⇒ (StudentEmotion,Sadness → Joy)[Support :34.0, Confidence : 85.0%]	Increase the number of team members in group projects	0.40	0.4

The explanation for choosing the cost and feasibility is shown below:

1. AR1 for Table 4 - To implement the suggested Action Rule, we must implement the Meta Actions, and we estimate that this will cost 0.80, because we need to train the student psychologically.
2. AR2 for Table 4- To implement the suggested Action Rule, we need to implement the Meta actions, the estimated cost is predicted as 0.70. We need to have diversified students, which the college will hire.
3. AR3 for Table 4 - To implement the suggested Action Rule, we need to implement the Meta actions, the estimated cost is predicted as 0.40. It requires to have more number of team members in the group.

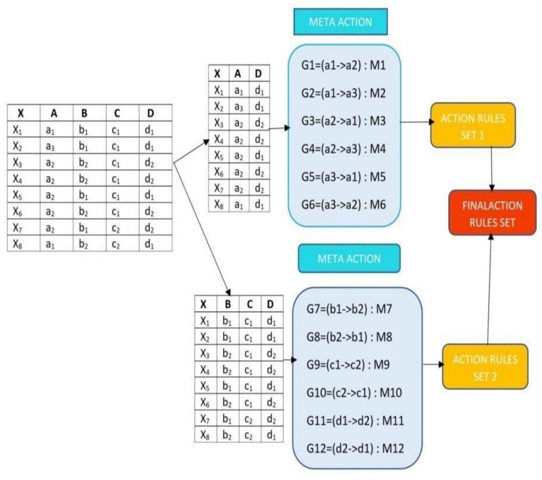

Figure 4. Vertical Data Split with Meta Action for Table 1.

6.2. Experiment 2 - Vertical Data Split generating Meta Action with NPS Data

To test our methods, we use the dataset: the Net Promoter Score data [24]. We used a sample of the Net Promoter Score dataset [24] for our experiments. The decision attribute in the dataset is PromoterStatus which labels each customer as either a promoter, a passive or detractor. The decision problem here is to enhance customer satisfaction / loyalty, as measured by Net Promoter Score. The goal of applying action rules to solve the problem is to find minimal sets of actions so that to "reclassify" customers from "Detractor" to "Promoter" and improve NPS. Figure 6 is showing how we generate Meta Action. We have created an extension to our previous work by generating Cost and feasibility for the Action Rules as shown in Table 5.

Experts provided the Meta Actions, but we have some probable predictions of Meta Actions.

The explanation for choosing the cost and feasibility is shown below:

1. For AR1 for Table 5 - To implement the suggested Action Rule, we need to implement the Meta Actions, which will cost 0.50, as the process involves UI redesign, frontend development, accessibility improvements, performance optimization, and QA user testing.
2. For AR2 for Table 5 - To implement the suggested Action Rule, we need to implement the Meta Actions, which will cost 0.30, as the process involves on-site and online training, workshops, simulation exercises, and assessments to train the professionals.
3. For AR3 for Table 5 - To implement the suggested Action Rule, we need to implement the Meta Actions, which will cost 0.60, as the process is slightly more complex, as we have to create a portal for the order system through which our customers will be connected to the inventory. It requires analyzing existing orders, building software and teams based on inventory and order management, storing order details in a database, developing APIs to sync real-time orders, reflecting orders in the UI, and testing end-to-end (E2E) flows.
4. For AR4 for Table 5 - To implement the suggested Action Rule, we must implement the Meta Actions, which will cost 0.70, as the process involves introducing new features or processes. The product requires the development and manufacturing of

new features, compatibility with existing features, regression integration, end-to-end testing, and training the team and customers about the new implementation.

5. For AR5 for Table 5 - To implement the suggested Action Rule, we need to implement the Meta Actions, which will cost $0, as the process requires no changes.

Table 5. Action Rules of Net Promoter Score datasets:17 California part1ActionRules to change promoter status from detractor to promoter

Experiments	**Meta-Action**	**Cost (0-1)**	**Feasibility (0-1)**
ARN1:(BenchmarkPartsEaseofCompletingPartsOrder,5→9)=⇒ (PromoterStatus,Detractor →Promoter)[Support : −4.0,Confidence : −52.72%]	Enhance website design to make it user friendly	0.50	0.2
ARN2:(BenchmarkPartsEaseofCompletingPartsOrder,5→8)=⇒ (PromoterStatus,Detractor → Promoter)[Support : −2.0,Confidence : −61.53%]	Training Company Professionals to respond to customer better	0.30	0.5
ARN3 : (BenchmarkPartsHowOrdersArePlaced,2→3)∧(ChannelType, ConstructionAl l→ConstructionAll) ∧(SurveyType,Parts→Parts)∧(BenchmarkPartsPromptNotificationofBackOrders,7→9)∧ (BenchmarkPartsTimeitTooktoPlaceOrder,8→10)⇒(PromoterStatus,Detractor→Promoter)[Support : −2.0,Confidence : −95.65%]	Connect customer order system with the inventory	0.60	0.6
ARN4: (BenchmarkAllOverallSatisfaction,7→7)⇒ (PromoterStatus,Detractor→Promoter) [Support : −3.0,Confidence : − 83.33%]	Improving the quality of the product to make the client happier	0.70	0.8
ARN5:(ClientName,HoltofCalifornia→HoltofCalifornia) =⇒ (PromoterStatus,Detractor →Promoter)[Support : −2.0,Confidence : − 53.43%]	Here the client is same	0	0

7. CONCLUSION

This paper introduced a novel and intelligent approach to data partitioning that significantly enhances both efficiency and the discovery of actionable knowledge. Our method, which utilizes a partition threshold, Rho (ρ), to horizontally and vertically divide large datasets, effectively reduces computational complexity and expense. By generating Meta Actions from the intersection of high-support sets, we have demonstrated a powerful mechanism for producing valuable information granules. The application of this method across diverse domains, including the Student Survey Dataset and the NPS (Net Promoter Score – Business), highlights its adaptability and potential to provide tangible insights for decision-making in both business and education.

A key benefit of our approach is its superior processing time when compared to existing methods, making it a more practical solution for large-scale data analysis. Furthermore, the analysis of the cost and feasibility of each Meta Action empowers users to make informed, goal-oriented decisions. Using the Monte Carlo method on our Action Rules generated by 50 randomly generated data, we were able to determine the frequency of the most common rules, identifying them as the most effective. While we acknowledge that there may be a slight decrease in the quality of the rules generated, the overall efficiency and practical application of our method make it a powerful tool for data analysis and decision-making.

8. FUTURE WORK

Future work will focus on refining the algorithm to further minimize the trade-off between rule quality and computational efficiency. This will involve exploring advanced heuristics, such as an adaptive ρ value that dynamically adjusts based on data characteristics, to ensure that the actionable knowledge derived from our method remains both highly relevant and robust. We also plan to evaluate our approach to a wider range of real-world, high-volume datasets to validate its scalability and performance across different industries. Additionally, we will investigate the integration of our method with other Action Rule methods like GA based Action Rule mining method extracting action rules directly from data which can result in more valuable action rules, machine learning method like SVM that automate the selection and application of Meta Actions, further solidifying its value in the field of data analytics.

9. REFERENCES

[1] M. Khder, I. Abu-AlSondos, and Y. Bahar, "The impact of implementing data mining in business intelligence," in International Journal of Entrepreneurship, vol. 25, pp. 1–9, 05 2021.

[2] J. Kuang, A. Daniel, J. Johnston, and Z. W. Ras, "Hierarchically structured recommender system for improving nps of a company," in International Conference on Rough Sets and Current Trends in Computing. Springer, 2014. pp. 347–35.

[3] J. Ranganathan, A. S. Irudayaraj, A. Bagavathi, and A. A. Tzacheva, "Actionable pattern discovery for sentiment analysis on twitter data in clustered environment," Journal of Intelligent & Fuzzy Systems, no. Preprint, pp. 1–15, 2018.

[4] Z. W. Ras and A. A. Tzacheva, "In search for action rules of the lowest cost," in Monitoring, Security, and Rescue Techniques in Multiagent Systems. Springer, 2005, pp. 261–272.

[5] P. Su, D. Li, and K. Su, "An expected utility-based approach for mining action rules," in Proceedings of the ACM SIGKDD Workshop on Intelligence and Security Informatics, ser. ISI-KDD'12. New York, NY, USA: ACM, 2012, pp. 9:1–9:4. [Online].Available: http://doi.acm.org/10.1145/2331791.2331800

[6] A. Tzacheva and Z. Ras, "Action rules mining." in International Journal of Intelligent System, vol. 20, pp. 719–736, 01 2005.

[7] L.-S. Tsay and Z. W. Ras, "Action rules discovery: system dear2, method and experiments," Journal of Experimental & Theoretical Artificial Intelligence, vol. 17, no. 1-2, pp. 119–128, 2005.

[8] Z.W.Ras, E. Wyrzykowska, and H. Wasyluk, "Aras: Action rules discovery based on agglomerative strategy," in International Workshop on Mining Complex Data. Springer, 2007, pp. 196–208.

[9] Z. W. Ras, A. Dardzinska, L.-S. Tsay, and H. Wasyluk, "Association action rules," in Data Mining Workshops, 2008. ICDMW'08. IEEE International Conference on. IEEE, 2008, pp. 283–290.

[10] S. Im and Z. W. Ras, "Action rule extraction from a decision table: Ared," in International Symposium on Methodologies for Intelligent Systems. Springer, 2008, pp. 160–168.

[11] S. R. A.A. Tzacheva, C.C. Sankar and R. Shankar, "Support confidence and utility of action rules triggered by meta-actions," in proceedings of 2016 IEEE International Conference on Knowledge Engineering and Applications, ser. ICKEA 2016. IEEE Computer Society, 2016.

[12] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," Communications of the ACM, vol. 51, no. 1, pp. 107–113, 2008.

[13] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M. J. Franklin, S. Shenker, and I. Stoica, "Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing," in Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation, ser. NSDI'12. Berkeley, CA, USA: USENIX Association, 2012, pp. 2–2.

[14] A. Bagavathi, P. Mummoju, K. Tarnowska, A. A. Tzacheva, and Z.W. Ras, "SARGS method for distributed actionable pattern mining using spark," in IEEE International Conference on Big Data (Big Data), vol.12, pp. 4272–4281, 2017.

[15] J. Kuang, A. Daniel, J. Johnston, and Z. W. Ras, "Hierarchically structured recommender system for improving NPS of a company," in international conference on Rough Sets and Current Trends in Computing, C. Cornelis, M. Kryszkiewicz, D. Slzak, E. M. Ruiz, R. Bello, and L. Shang, Eds.´ Cham: Springer International Publishing, pp. 347–357, 2014.

[16] A. A. Tzacheva and A. Easwaran, "Modified hybrid scalable action rule mining for emotion detection in student survey data," in International Conference on Data Mining, Big Data, Database and Data Technologies, pp. 13–19, 01 2022.

[17] K.K. Ramachandran, "Natural language processing for legal documentation in indian languages," International Journal of Business and Data Analytics (IJBDA), vol. 1, pp. 1–15, 01 2024.

[18] S.B. Vinay, "Identifying research trends using text mining techniques: A systematic review," International Journal of Data Mining and Knowledge Discovery (IJDMKD), vol. 1, pp. 1–11, 01 2024.

[19] S.B. Vinay, "Natural language processing for Legal Documentation in indian languages," International Journal of Natural Language Processing (IJNLP), 2024, vol. 2, pp. 1–11.

[20] Z. W. Ras, K. A. Tarnowska, J. Kuang, L. Daniel, and D. Fowler, "User friendly NPS-based recommender system for driving business revenue", in "International Joint Conference on Rough Sets", pp. 34–48, 06 2017.

[21] M. Karim and R. M. Rahman, "Decision tree and naive bayes algorithm for classification and generation of actionable knowledge for direct marketing," Journal of Software Engineering and Applications, vol. 6, no. 04, p. 196, 2013.

[22] Z. Cui, W. Chen, Y. He, and Y. Chen, "Optimal action extraction for random forests and boosted trees," in Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015, pp. 179–188.

[23] Y. Hu, J. Du, X. Zhang, X. Hao, E. Ngai, M. Fan, and M. Liu, "An integrative framework for intelligent software project risk planning," Decision Support Systems, vol. 55, no. 4, pp. 927–937, 2013.

[24] Y. Hu, B. Feng, X. Mo, X. Zhang, E. Ngai, M. Fan, and M. Liu, "Cost-sensitive and ensemble based prediction model for outsourced software project risk prediction," Decision Support Systems, vol. 72, pp. 11–23, 2015.

[25] A. Bagavathi, V. Rao, and A. A. Tzacheva, "Data distribution method for scalable actionable pattern mining," in Proceedings of the First International Conference on Data Science, E-learning and Information Systems. ACM, 2018, p. 3.

[26] J. Kuang, Z. W. Ras, and A. Daniel, "Personalized meta-action mining for nps improvement," in Foundations of Intelligent Systems, F. Esposito, O. Pivert, M.-S. Hacid, Z. W. Ras, and S. Ferilli, Eds. Cham: Springer International Publishing, 2015, pp. 79–87.

[27] A. Bagavathi, A. Tripathi, A. A. Tzacheva, and Z. W. Ras, "Actionable pattern mining - a scalable data distribution method based on information granules," in 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), 2018, pp. 32–39.

[28] S. Kusal, S. Patil, K. Kotecha, R. Aluvalu, and V. Varadarajan, " AI based emotion detection for textual big data: Techniques and contribution," Big Data and Cognitive Computing, 2021, Vol. 5, issue 3, 43 Doi: 10.3390/bdcc5030043.

[29] S. Ragunathan, "Emotional Detection from Text Using NLP (Natural Language Processing) With Prediction Probability," 2024

[30] S.Hashemi, & P. Shamsinejad, Ga2rm: A ga-based action rule mining method. International Journal of Computational Intelligence and Applications,2021, 20(02), 2150012.

[31] Z. W. Ras and A. Wieczorkowska, "Action-rules: How to increase profit of a company," in European Conference on Principles of Data Mining and Knowledge Discovery. Springer, 2000, pp. 587– 592.