

Subgroup Discovery Algorithms: A Survey and Empirical Evaluation

Sumyea Helal

School of Information Technology and Mathematical Sciences, University of South Australia, Adelaide, SA5001, Australia

E-mail: sumyea.helal@mymail.unisa.edu.au

Received February 12, 2015; revised March 19, 2016.

Abstract Subgroup discovery is a data mining technique that discovers interesting associations among different variables with respect to a property of interest. Existing subgroup discovery methods employ different strategies for searching, pruning and ranking subgroups. It is very crucial to learn which features of a subgroup discovery algorithm should be considered for generating quality subgroups. In this regard, a number of reviews have been conducted on subgroup discovery. Although they provide a broad overview on some popular subgroup discovery methods, they employ few datasets and measures for subgroup evaluation. In the light of the existing measures, the subgroups cannot be appraised from all perspectives. Our work performs an extensive analysis on some popular subgroup discovery methods by using a wide range of datasets and by defining new measures for subgroup evaluation. The analysis result will help with understanding the major subgroup discovery methods, uncovering the gaps for further improvement and selecting the suitable category of algorithms for specific application domains.

Keywords subgroup discovery, searching, pruning, measure, evaluation

1 Introduction

Data mining aims to discover knowledge from databases and it has been defined as the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data^[1]. Data mining techniques can be divided into descriptive and predictive induction^[2]. Predictive induction extracts knowledge with an intent to predict the class value of unknown examples whereas descriptive induction aims to discover interesting knowledge from data as a form of patterns.

Subgroup discovery (SD) is a descriptive induction technique that extracts interesting relations among different variables with respect to a special property of interest known as target variable. For example, an institute might be interested in knowing what are the circumstances that lead students having a higher failure therefore to take necessary steps for reducing the failure rate.

Subgroup discovery possesses a wide range of real-world applications. Its applications in medical domain include the detection of coronary heart disease^[3] as well as brain ischaemia^[4–5]. It has numerous contribu-

tions in the marketing domain for analyzing financial^[6] and commercial^[7] data. It is applicable in other areas like learning^[8], mining of census data^[9] and vegetation data^[10] as well.

A number of subgroup discovery algorithms^[11–15] have been developed so far. They differ in the strategies for searching candidates and in the measures for ranking interesting subgroups. In view of the significant number of existing subgroup discovery algorithms, the need arises to examine the features of a subgroup discovery algorithm to generate quality subgroups. To meet up this requirement, a number of reviews^[2,16] have already been conducted on subgroup discovery. Although these studies provide a broad overview on some subgroup discovery methods by analyzing the main properties, models and problems solved by the methods, most of them are confined to only theoretical analysis. An application study^[8] has conducted experiments on several subgroup discovery methods. Though this work provides an in-depth understanding of some subgroup discovery algorithms, the experiments are performed by employing only a single dataset and the discovered subgroups are evaluated with respect to few measures.

Therefore, in this paper, we assess the performance of several fundamental algorithms found in the literature by conducting extensive experiments on them.

The contributions of this work include:

1) providing an in-depth understanding of different subgroup discovery algorithms by conducting an extensive study on them;

2) evaluating existing algorithms, finding out their gaps by performing an empirical analysis on them, proposing three new measures, and also using some other measures proposed by some popular SD approaches for assessing their performance;

3) providing some suggestions about which features should be considered for domain specific subgroup discovery.

To achieve the above goals, we conduct an extensive survey of some popular subgroup discovery methods, define new measures for subgroup evaluation, and employ a wide range of datasets for evaluating the performance of the algorithms. We then conduct an in-depth analysis of the results and provide recommendations accordingly. The resulting analysis from the first two objectives can serve as a broad area for the researchers to develop a better subgroup discovery method. The outcome of the third objective will aid the data mining practitioners to choose a suitable method for applying within a specific domain.

The paper is organized as follows. Section 2 introduces some background knowledge on subgroup discovery and the methodology. Section 3 reviews the fundamental subgroup discovery algorithms and their potential applications. Section 4 describes the evaluation measures employed in the experiments. Section 5 presents an empirical analysis and interpretation of the results, followed by Section 6 as concluding remarks.

2 Background

The following subsections describe the concept of subgroup discovery with a simple illustrative example and the methodology of subgroup generation.

2.1 What Is Subgroup Discovery

The notion of subgroup discovery has been defined by Klösigen^[17] and Wrobel^[18] as:

“Given a population of individuals and a property of those individuals that we are interested in, find population subgroups that are statistically ‘most interesting’, for example, are as large as possible and have the most

unusual statistical (distributional) characteristics with respect to the property of interest.”

Subgroup discovery aims to extract relations among different variables with respect to the property of interest. These relations are represented in the form of rules represented as follows:

$$\text{Condition} \rightarrow \text{Target},$$

where *Target* is a value for the property of interest and *Condition* is a conjunction of attribute-value pairs representing relations characterized by the value of *Target*.

Some possible relations among the above attributes shown in Table 1 can be as follows:

S1: (Salary > 80K AND Education = University) → Approved = Yes;

S2: (Salary < 50K AND Children = Yes AND Education = Secondary) → Approved = No.

Table 1. Example Dataset

Salary	Children	Education	Approved
< 50K	Yes	Secondary	No
> 80K	No	University	Yes
> 80K	Yes	University	Yes
50K~80K	No	Secondary	No
50K~80K	No	University	Yes
< 50K	Yes	Secondary	No
50K~80K	Yes	Secondary	No

The first subgroup represents that people with higher salary and education level have a higher probability for loan approval with respect to people with lower salary and education level. The second subgroup with lower salary and education level and having children has a very high probability of loan disapproval.

2.2 Descriptive Versus Predictive Rule Discovery

Data mining is the process of discovering knowledge from databases in form of patterns or rules. Data mining methods can be classified into two groups according to their objectives — predictive and descriptive induction^[7]. The predictive induction methods such as classification aims to predict or classify the unknown object from the discovered knowledge while the descriptive induction methods, e.g., association rule mining or clustering, reveal interesting knowledge from the data.

Subgroup discovery is represented as a task at the intersection of descriptive and predictive induction^[17].

However, subgroup discovery differs from descriptive induction techniques as they attempt to extract relations between unlabelled objects while subgroup discovery searches for relations with respect to a special property of interest.

Although the predictive induction methods deal with labelled data, they are different from subgroup discovery in terms of the purpose of knowledge discovery from databases. The goal of classification is to generate a model for each class that contains rules representing class characteristics regarding the descriptions of training examples. This model is used to predict the class of unknown object. In contrast, subgroup discovery attempts to discover interesting relations with respect to the property of interest.

2.3 Subgroup Discovery Methodology

As illustrated in Fig.1, a subgroup discovery algorithm consists of three major phases for extracting subgroups — candidate subgroup generation, pruning and post-processing. In the following, we describe these steps in detail.

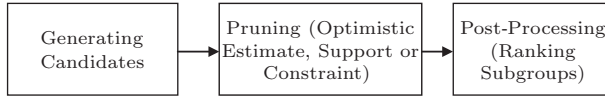


Fig.1. Methodology for subgroup discovery.

2.3.1 Candidate Subgroup Generation

Each subgroup discovery algorithm uses a specific strategy to search for the candidate subgroups. Such strategy is very essential for extracting subgroups as the volume of search space is exponential with respect to the number of attributes and their values. The search space is traversed by starting with simple descriptions and processing them in a more general to specific manner by adding up more attribute-value pairs. Different search strategies have been employed so far for subgroup discovery; among them, the most widely used strategies are exhaustive search, beam search and genetic algorithm (GA) based search.

Exhaustive Search. Exhaustive search is a very popular problem solving technique which generates all possible candidates and verifies whether each candidate satisfies some specific constraints. When exhaustive search is feasible, depth first search is generally employed. Depending on the possibility,

the anti-monotone property^[19] or optimistic estimate function^[20] is used to restrict the search space^[14]. The cost of this type of search is proportional to the number of generated candidates; hence exhaustive search is not affordable when the search space is too huge.

Beam Search. When exhaustive search is not possible, beam search is the commonly used heuristic substitute. It implements a level-wise top-down approach for extracting subgroups. In beam search, only a predefined number (known as beam width bw) of the best partial solutions are taken as candidates. At each level, the bw highest ranked candidates are generated according to the quality. The initial candidate is generated from empty subgroup description. Beam search restricts the memory usage by exploring part of the search space; while it does not guarantee solution at the end.

Genetic Algorithm. Genetic algorithm (GA) is a search heuristic that follows the process of natural evolution; hence the methods implementing this search heuristic are known as evolutionary methods. This heuristic is used to extract solutions to different optimization and search processes. According to GA, each solution is composed of several variables and equipped with a fitness score. The solutions with higher fitness values are given the opportunity to evolve. GA provides significant benefits over other search strategies, e.g., linear programming, depth-first search and breath-first search.

2.3.2 Pruning

In the second phase, a subgroup discovery algorithm needs to employ a pruning scheme for selecting only the significant candidates. A number of pruning strategies are used by different methods. The major types include minimum support or coverage pruning, optimistic estimate pruning, and constraint pruning. Minimum support pruning allows a subgroup discovery method to select only those candidates that have a minimum occurrence frequency in the dataset. A number of popular methods^[11-13] implement this pruning technique. Some subgroup discovery methods^[20-21] implement the optimistic estimate as a pruning criterion which has been defined in [18] as follows

Definition 1 (Optimistic Estimate). *An optimistic estimate $oe(s)$ for a given quality function q is a function that satisfies the following: \forall subgroups s, s' . $s' \succ s \implies oe(s) \geq q(s')$. Here s' represents the refinement of subgroup s such that the subgroup description $sd = \{i_1, i_2, \dots, i_n\}$ of s is a subset of the subgroup description $sd' = \{i'_1, i'_2, \dots, i'_n\}$ of s' .*

Quality constraint is also one of the widely used pruning strategies for some subgroup discovery methods^[22-23]. These methods prune the candidates that have a lower quality than a user-specified threshold value.

2.3.3 Post-Processing

The final phase of subgroup discovery algorithm implements a quality measure in the purpose of ranking subgroups. These measures are very vital for evaluating subgroups as the interest attained directly relies on them. It can be defined as follows.

Definition 2 (Quality Measure). *A quality measure is a function φ which assigns a numeric value to a subgroup $S \subseteq D$.*

Different quality measures are used by different subgroup discovery algorithms. Among them, unusualness^[24] and Piatetsky-Shapiro^[17] are the most popular ones.

The pruning techniques and the quality measures used by different methods are summarized in Table 2.

3 Subgroup Discovery

A number of subgroup discovery algorithms have been proposed till now. The following subsections discuss some popular beam search based methods, exhaustive search based methods, and genetic algorithm based methods and their applications in different domains.

3.1 Existing Approaches

Existing subgroup discovery approaches can be broadly classified into three major categories according to the strategy for searching the candidates — beam search based approaches, exhaustive search based approaches, and genetic algorithm based approaches.

3.1.1 Beam Search Based Algorithms

These algorithms generate a fixed number of candidates (specified by beam width parameter) during each iteration. At each level, a refinement operator generates candidates for the next level by searching from the candidates contained in the beam. Though beam search is faster as compared with other search strategies, it only explores part of the search space and does not guarantee an optimal solution. A number of beam search based algorithms have been developed till now. The most popular ones are described below.

SubgroupMiner^[25]. This subgroup discovery system provides the main ingredients for description languages, search strategies and evaluation of hypothesis, visualization and causal subgroups. This method implements binomial test^[17] as quality measure. It can work on both numeric and nominal target attributes.

SD^[11]. SD is a heuristic beam search based algorithm in which the final subgroup set is selected according to the opinion of an expert instead of simply using a measure for subgroup searching and selecting. The algorithm generates subgroups according to a fixed beam

Table 2. Pruning Techniques and Quality Measures Used by Different Algorithms

Search Strategy	Method	Pruning Technique	Quality Measure
Beam	SubgroupMiner	Minimum support	Binomial test
	SD	Minimum support	Generalization quotient
	CN2-SD	Constraint	Unusualness
	RSD	Minimum support	Unusualness, significance or coverage
	DSSD	Minimum coverage	Unusualness, Chi-squared, mean test, or WKL
Exhaustive	EXPLORA	No pruning	Generality, redundancy, or simplicity
	MIDOS	Minimum support and optimistic estimate	Novelty or distributional unusualness
	APRIORI-SD	Minimum support	Unusualness
	SD-Map	Minimum support and coverage	Piatetsky-Shapiro, unusualness, or binomial test
	DpSubgroup	Tight optimistic estimate	Piatetsky-Shapiro, or Pearson's Chi-square
Evolutionary	MergeSD	Constraint pruning	Piatetsky-Shapiro
	SDIGA	No pruning	Confidence, support, unusualness, sensitivity, interest, significance
	MESDIF	No pruning	Confidence, support, unusualness, sensitivity, significance
	NMEEF-SD	No pruning	Confidence, support, unusualness, sensitivity, significance
	CGBA-SD	No pruning	Confidence, support

width. In each iteration of this algorithm, the subgroup description is updated by adding up new attribute values. Discovered subgroups must maintain a minimum frequency and should be relevant for acceptance.

CN2-SD^[23]. CN2-SD is a subgroup discovery algorithm. It is an extension of the popular classification rule induction algorithm CN2^[26]. Unlike the standard CN2 algorithm, it uses a modified unusualness^[24] for ranking subgroups. CN2-SD implements a different covering scheme as compared with the standard CN2 algorithm. In CN2, when a new rule is found, all the examples covered by the rule are removed to avoid generating the same rule again. In CN2-SD, the covered examples are not eliminated from the training dataset; rather a count is stored with each example indicating the frequency the example has been covered so far. If an example is covered by a rule, its weight decreases and its rule quality is measured by considering the new weight of this example.

RSD^[27]. This method extracts subgroups from a relational database by removing irrelevant features. It implements modified weighted covering and weighted WRAcc (Weighted Relative Accuracy) heuristic for extracting subgroups. It employs unusualness as the quality measure for ranking subgroups.

DSSD^[14]. DSSD is a beam search based non-redundant subgroup discovery algorithm. It implements three different heuristic selection strategies to manage redundancy. DSSD assumes that in a non-redundant subgroup set, all subgroup pairs should be different in 1) subgroup description, 2) subgroup cover, or 3) exceptional models. DSSD generates a large number of j subgroups in the initial phase. In the second phase, these j subgroups are refined by incorporating dominance pruning which removes the subgroup condition one by one if it does not decrease the quality of this subgroup. In the final phase, a subgroup selection strategy is used to select k best subgroups from the initially generated j subgroups. DSSD uses WRAcc, Chi-squared^[20], mean test^[17], or WKL (Weighted Kullback-Leibler Divergence)^[14] to rank the subgroups.

Different types of heuristic are used by different methods for extracting and evaluating subgroups. SubgroupMiner uses the binomial test for ranking subgroups, while SD employs generalization quotient for evaluating them. Both CN2-SD and RSD implement modified covering scheme and modified WRAcc for generating and ranking subgroups, although CN2-SD is a propositional subgroup discovery algorithm and RSD is a relational subgroup discovery algorithm. In-

corporating the weighted covering scheme allows these methods to manage redundancy to some extent and increase diversity in the generated subgroups. DSSD implements three subgroup selection strategies for reducing the number of redundant subgroups.

3.1.2 Exhaustive Search Based Algorithms

These algorithms search for every possible candidate and prune the irrelevant subgroups according to some constraints to reduce the hypothesis space. This is a widely used search technique because of its ease of implementation; still it is not feasible to use when the number of candidate solution grows quickly with the problem size. Some of the widely used exhaustive search based subgroup discovery algorithms are described below.

EXPLORA^[17]. This is the first approach for subgroup discovery. It implements decision trees for generating subgroups. The interestingness of rules is evaluated by different statistical measures such as generality, simplicity.

MIDOS^[18]. This is the first subgroup discovery method that extracts interesting subgroups from multi-relational databases. For subgroup evaluation, this method uses a function that combines both the unusualness and the size of a subgroup. This approach implements both the minimum support pruning and the optimistic estimate pruning for selecting candidate subgroups.

APRIORI-SD^[12]. APRIORI-SD is an extension of the classification rule learning algorithm, APRIORI-C^[28]. Each subgroup must maintain a minimum support and confidence threshold to have their place in the final result set. Discovered subgroups are post-processed by using unusualness as the quality measure. All the positive examples covered by a rule are not removed from the training dataset, rather each time an example is covered, and its weight decreases. An example is removed only when its weight falls below a given threshold or when an example has been covered more than k times.

SD-Map^[13]. SD-Map is an exhaustive subgroup discovery method which is an extension of the popular Frequent Pattern (FP) Growth^[29] based association rule mining method. It selects only those subgroups that maintain a minimum support value within the dataset. It implements a depth-first search for candidate generation and also is able to handle missing values for different domains. It uses several quality functions such as Piatetsky-Shapiro^[17], unusualness.

DpSubgroup^[20]. This subgroup discovery algorithm implements an FP-tree based structure for generating subgroups. It uses tight optimistic estimate pruning for removing irrelevant subgroups. This algorithm incorporates the optimistic estimate for Piatetsky-Shapiro and Pearson's Chi-square test.

MergeSD^[22]. This algorithm employs a depth-first strategy for searching the candidate subgroups. It implements constraint-based pruning on the subgroups over overlapping intervals. This algorithm can work on the datasets having numeric attributes by pruning large part of search space by exploiting bounds among related numeric subgroup descriptions.

MIDOS is an adaption of the EXPLORA approach which aims to extract statistically unusual subgroups in multi-relational databases. APRIORI-SD, SD-Map, DpSubgroup, and MergeSD are the extensions of different association algorithms. APRIORI-SD is an adaption for subgroup discovery from the classification rule learning algorithm APRIORI-C^[28] which is a modification of the APRIORI^[30] association rule learning algorithm. It employs a candidate generation and test strategy. SD-Map, DpSubgroup, and MergeSD are the extensions of FP-growth^[29] based methods which use divide-and-conquer strategy for searching the hypothesis space.

3.1.3 Genetic Algorithm Based Approaches

Genetic algorithm based approaches have been gaining attention in recent days. This type of search heuristic follows the process of natural evolution such as inheritance, mutation, selection and crossover. Genetic algorithms belong to the larger class of evolutionary algorithms which usually tackle the optimization problems. Some of the most recent evolutionary algorithms are given below.

SDIGA^[15]. This is an evolutionary fuzzy rule induction algorithm. Fuzzy rules are implemented in disjunctive normal form (DNF) as description language for representing subgroups. DNF fuzzy rules allow all the variables to take multiple values and facilitate the discovery of more general rules. It evaluates the rules by performing a weighted average of some quality measures which may include confidence, support, and unusualness.

MESDIF^[31]. This is a multi-objective genetic algorithm for inducing fuzzy rules. This algorithm implements the SPEA2 (Strength Pareto Evolutionary Algorithm 2)^[32] approach and applies elitism for selecting rules. It can employ several quality measures to eval-

uate the generated subgroups. These measures include support, confidence, and unusualness.

NMEEF-SD^[33]. This is an evolutionary multi-objective fuzzy rule induction system. This algorithm incorporates the use of some special operators to generate simple and high quality subgroups. This algorithm is based on a non-dominated sorting approach NSGA-II (Non-Dominated Sorting Genetic Algorithm II)^[34]. For the extraction and evaluation of subgroups, this method may employ several quality measures, which may include confidence, support, and unusualness.

CGBA-SD^[35]. This algorithm represents a grammar guided genetic programming approach for subgroup discovery. According to this method, each rule is represented as a derivation tree that shows a solution represented using the language denoted by the grammar. The algorithm consists of the mechanisms to adapt the diversity of the population by self-adapting the probabilities of recombination and mutation.

The evolutionary algorithms developed so far implement a hybridization of genetic algorithm and fuzzy system, known as Genetic Fuzzy System (GFS). GFS provides new and useful tools for analyzing patterns and discovering useful knowledge.

3.2 Applications in Different Domains

Subgroup discovery possesses a wide range of applications in different fields where knowledge related to a specific target value has higher interest. This subsection describes some significant applications of subgroup discovery in different areas.

Medical Domain. In medical domain, subgroup discovery has been widely applied to detect the risk groups with coronary heart disease. In [3, 7, 36-37], some influencing factors (such as high cholesterol level, density of lipoprotein, and triglyceride) have been detected for the patients at high risk. All these studies implement SD algorithm for extracting the risk factors as the evaluation of such properties needs the involvement of domain experts for effectively searching the hypothesis space.

Several other studies^[4-5] have detected some principal and supporting factors of the patients having brain stroke. The goal of this type of work is to help these patients by diagnosing and preventing the disease. The significant subgroups are discovered by SD algorithm with the help of human expert with an intent to select a relatively small set of relevant hypotheses.

In this domain, another interesting work^[38] has been proposed, which discovers the subgroups of

patients visiting the psychiatric emergency department. This work implements an evolutionary algorithm SDIGA to discover the fuzzy rule that states the relationship among the arrival time of different patients at this unit. A fuzzy genetic algorithm has also been employed in [39] to address the problem of pathogenesis of acute sore throat condition in human.

Bioinformatics. Subgroup discovery has addressed different real-world problems in the bio-informatics domain. In [37, 40], relevant features are extracted by implementing the SD subgroup discovery algorithm for the detection of different cancer types. [41] employs an SD approach SD4TS (Subgroup Discovery for Test Selection) for breast cancer diagnosis. Some other studies^[42-43] have detected the groups of gene that are highly correlated with the class of lymphoblastic leukemia. They have employed the RSD algorithm for inducing subgroups that characterize the genes in terms of the knowledge extracted from gene ontologies and interactions. In another work^[44], the PET scan has been analyzed for extracting the patient groups that have similar features in brain metabolism. In this work, the RSD algorithm is implemented to extract the relationship and knowledge obtained from patients' database to identify who are unusual with respect to the images.

Marketing. Subgroup discovery also possesses significant applications in the marketing domain. The very first application^[6] in this domain is an analysis of financial market research with the help of the EXPLORA algorithm. In this application, data is collected by interviewing persons about their behavior in the market. This type of analysis may incur the missing data problem and need sufficient preprocessing of the collected data. A different type of market analysis has been conducted in [36], which discovers the subgroups of customers who are familiar with different market brands. The features are extracted by using SD algorithm. A similar type of analysis has been conducted in [7], but here the subgroups of the potential customers of different brands have been extracted by employing the CN2-SD algorithm.

Spatial Data Analysis. Spatial data analysis has also been achieved by subgroup discovery. In [9], the SubgroupMiner algorithm has been employed on UK census data to extract the information of enumeration districts with a high migration rate. Some influencing factors such as the low rate of households with two cars, the low rate of married people and the lower unemployment rate in some areas are considered for higher migration rate. In another work [25], mining from census

data has been conducted for extracting the possible effects on mortality by using SubgroupMiner algorithm. Spatial data has been analyzed in [10] containing 132 vegetation records which describe the cites of various plants. The main objective of this work is to evaluate subgroups that are in favor of the existence of a plant species.

Other Domains. The popularity of the web-based education system has been on the hype in recent days. Mining educational data possesses a wide range of objectives which commonly include analyzing students' behavior, detecting vulnerable students, recommending relevant books and so on. Different subgroup discovery algorithms have been compared in [8] by using e-learning data obtained from the Moodle e-learning system from the University of Cordoba. The aim is to retrieve knowledge from usage data and improve students' performance by using it.

A study on traffic accidents has been conducted in [45-46]. The first work conducted a comparison of the extracted subgroups between the standard CN2 and the CN2-SD method, and the second paper compares the subgroups generated by SubgroupMiner and CN2-SD.

From the above discussion, it is observed that different algorithms have been implemented in individual application domain. Hence each algorithm has their own methodology for subgroup extraction. Since each application area possesses some requirements with respect to the discovered subgroups, it is very crucial to learn the features that should be considered for a domain specific subgroup discovery.

4 Evaluation Measures

In this section, the definitions of different evaluation measures are given. Several existing measures are used in the experiments in this paper for evaluating the subgroups generated by different algorithms. At the same time, three new measures have been proposed by this work — overall coverage, overall support and redundancy. Each of these measures has different objectives in terms of evaluating the subgroups.

Preliminaries. Let D be a dataset. All the examples to be analyzed are represented by the set of k description attributes A (where $A = \{A_1, A_2, \dots, A_k\}$) and a single class attribute c . Each attribute A_i has a domain of possible values $Dom(A_i)$. Our dataset D is now a collection of N examples over the set of attributes (A and c). Subgroup S is a set of description attribute-value pairs contained in D such that

$S \subseteq \{a_1, a_2, \dots, a_m | a_i \in \text{Dom}(A_i)\}$. φ is a quality measure that assigns a numeric value to subgroup S such that $S^c \subseteq D^c$.

$\text{coverset}(S)$ is the set of examples in D containing S . $\text{Cov}(S)$ represents the fraction of examples containing $\text{coverset}(S)$. $\text{supportset}(S, c)$ is the set of examples in D containing both S and c . $\text{Sup}(S, c)$ is the fraction of examples containing $\text{supportset}(S, c)$. class represents the set of examples containing the target attribute.

4.1 Measures of Complexity

These measures belong to the understandability of the knowledge extracted from the subgroups. These types of measures are as follows.

Number of Subgroups. It measures the number of discovered subgroups. In a beam search based method, the number of discovered subgroups is restricted by beam width. In a top- k method, the number of generated subgroups is limited by the value of k .

Length of Subgroups. It measures the number of variables contained in a subgroup. Subgroups with larger number of attributes are more specific to a particular target class.

4.2 Measures of Generality

These measures are used to quantify the generality of subgroups according to the individual patterns of interest covered^[16]. Within this group, the following measures can be found.

Definition 3 (Average Coverage). *It measures the percentage of examples covered on average by a single subgroup^[7]. The average coverage is computed as*

$$\text{AvCov} = \frac{1}{n_s} \sum_{i=1}^{n_s} \text{Cov}(S_i),$$

where n_s is the number of subgroups of the induced subgroup set.

Definition 4 (Overall Coverage). *It measures the fraction of examples covered collectively by a subgroup set. Overall coverage is given by the following equation.*

$$\text{COV} = \frac{|\bigcup_{i=1}^{n_s} \text{coverset}(S_i)|}{N}.$$

Although the average coverage measures the generality of a subgroup within the dataset, measuring overall coverage is very crucial. The average coverage measures the amount of examples to be covered on average by a single subgroup whereas overall coverage gives us an

idea about what proportion of examples have been covered by the discovered subgroup set. Hence it truly measures the generality of a subgroup set by counting both the target and the non-target class examples which belong to the subgroup of a rule set.

Definition 5 (Average Support). *The support of a subgroup measures the fraction of correctly classified examples covered by that subgroup^[12]. The average support of a single subgroup is given by the following equation,*

$$\text{AvSup} = \frac{1}{n_s} \sum_{i=1}^{n_s} \text{Sup}(S_i, c).$$

Definition 6 (Overall Support). *It measures the percentage of target examples covered by a subgroup set. Overall support can be represented as follows,*

$$\text{SUP} = \frac{|\bigcup_{i=1}^{n_s} \text{supportset}(S_i, c)|}{N}.$$

Overall support of a subgroup set measures the proportion of the target examples covered by the discovered subgroup set. It measures the degree of specificity of the subgroup set to a particular target class which cannot be achieved by evaluating rules using average support.

4.3 Measures of Precision

This category of measure evaluates the precision of the subgroups. The following measure quantifies the precision of the discovered subgroups.

Definition 7 (Average Confidence). *It measures the average frequency of target examples among those satisfying only the antecedent^[47]. The average confidence of a subgroup is given by the following equation,*

$$\text{AvConf} = \frac{1}{n_s} \sum_{i=1}^{n_s} \frac{|\text{supportset}(S_i, c)|}{|\text{coverset}(S_i)|}.$$

4.4 Measures of Interest

This measure is used for selecting and ordering interesting subgroups. A measure for evaluating the interestingness of the subgroups is as follows.

Definition 8 (Significance). *This measure indicates the level of significance of a finding. Average significance is computed in terms of the likelihood ratio of a subgroup^[16-17]. The average significance of a subgroup is as follows,*

$$\text{AvSig} = \frac{1}{n_s} \sum_{i=1}^{n_s} 2 \sum_{k=1}^{n_c} |\text{supportset}(S_i, c)|$$

$$\log \frac{|supportset(S_i, c)|}{|class_k| \cdot Cov(S_i)},$$

where n_c represents the value of target class variable.

4.5 Measure of Quality

This measure is used to evaluate the quality of the discovered subgroups. The quality measure used in the experiments is given below.

Definition 9 (Unusualness). *This measure is identified as the weighted relative accuracy WRAcc of a subgroup^[48]. The unusualness of a subgroup can be considered as a tradeoff between the coverage and the accuracy gain of a subgroup^[7]. The average unusualness of a subgroup is given by the following equation:*

$$AvWRAcc = \frac{1}{n_s} \sum_{i=1}^{n_s} Cov(S_i) \left(\frac{|supportset(S_i, c)|}{|coverset(S_i)|} - \frac{|class|}{N} \right).$$

4.6 Measure of Redundancy

This measure assesses the preciseness of an algorithm by calculating to what extent a rule set contains extraneous information. This measure is defined as follows.

Definition 10 (Redundancy). *It measures the fraction of redundant subgroups to the discovered subgroups. A subgroup S_k is redundant if there is another subgroup S_j such that*

- 1) $S_j \subseteq S_k$, and
- 2) $\varphi(S_j) \geq \varphi(S_k)$.

The percentage of redundancy is given by,

$$Redundancy(\%) = \frac{n_{red}}{n_s} \times 100,$$

where n_{red} is the number of redundant subgroups.

As an example of redundancy,

$S_k = \{male, smoking, 40 \sim 45 \rightarrow disease\}$,

$S_j = \{male, smoking \rightarrow disease\}$.

When S_j has equal or higher quality than S_k , which denotes that the conditions of S_j are sufficient to be classified as disease, S_k is redundant.

5 Experiments

This section describes the datasets employed in these experiments, the implementation, parameter settings, and the results, including efficiency, the scalability of some popular methods, and the evaluation of the discovered subgroups by each of these methods.

5.1 Datasets

In the experiments, we used 13 different datasets from UCI machine learning repository^[49]. The datasets Hypothyroid and Sick are located at the directory of Thyroid Disease dataset. The numeric attributes have been removed from the Adult and Census-Income dataset while they are discretized in the remaining datasets by using MLC++^[50]. A brief description of the datasets can be found in Table 3. The minor class was used as the property of interest.

Table 3. Datasets Used in the Experiments

Number	Dataset	Number of Instances	Number of Attributes	Minor Class (%)
1	Adult	48 842	8	23.9
2	Breast Cancer Wisconsin	699	9	34.5
3	Census Income	250 000	33	6.2
4	Diabetes	768	8	34.0
5	German Credit	1 000	15	30.0
6	Heart	270	13	44.0
7	Hypothyroid	3 163	23	4.8
8	Iris	150	4	34.0
9	Kr-vs-kp	3 196	36	47.8
10	Mushroom	8 124	22	48.2
11	Sick	2 800	26	6.1
12	Tic-Tac-Toe	958	9	34.7
13	Vote	435	16	38.6

5.2 Implementations

All the experiments have been conducted on a computer with 8 processors (i7, 3.40 GHz), 16 GB RAM, and 64-bit windows operating system. For SD, CN2-SD and APRIORI-SD, we used the implementation by Orange data mining software tool^[51]; for SD-Map by VIKAMINE^[52], DSSD by DSSD software tools^[14], and NMEEF-SD has been implemented by KEEL^[53].

5.3 Parameter Settings

The minimum coverage count is 10. The minimum support is 1%. The maximum depth refers to the number of description attribute-value pairs of a subgroup which has been set to 5. Beam width is set as 100. For the top- k methods, k is fixed to 100.

5.4 Experimental Results

The experiments compare five subgroup discovery methods, SD, CN2-SD, APRIORI-SD, SD-Map, and DSSD, from each of the sub-categories. The experiments compare the performance of the methods by 1)

measuring their efficiency and scalability and 2) evaluating the subgroups discovered by them.

5.4.1 Efficiency

The efficiency of the algorithms has been measured in terms of the execution time (in second) and the peak memory usage (in MB). According to these two measuring parameters, DSSD is highly efficient as it takes less time and memory for generating and storing subgroups. The maximum allowed execution time for these experiments is two hours. The execution time and memory usage by different algorithms can be viewed from Table 4 and Table 5. In the two tables, “-” represents “out of memory” and “no” represents “no subgroup returned” within two hours.

Datasets hypothyroid and Kr-vs-kp contain nearly equal instances, but from Table 4, it can be seen that Kr-vs-kp takes more time to generate subgroups and also from Table 5, it is seen that it has higher memory usage. The reason is that the Kr-vs-kp dataset possesses a large number of attributes; hence it takes more time and space to generate and store the candidates. A similar observation can be followed from the result of the Mushroom and Sick datasets. The experiments

show that the average time and memory consumed with the Mushroom dataset is lower as compared with that of the Sick dataset, although the former dataset possesses a higher number of instances as compared with the latter dataset.

5.4.2 Scalability

The scalability of the algorithms is evaluated in terms of the execution time and peak memory usage with different data sizes. From Fig.2, it is observed that only SD-Map extracts subgroups within two hours with 100K samples while the other methods discover subgroups with up to 50K instances within two hours. It is seen from Fig.2 and Fig.3 that DSSD is highly scalable. NMEEF-SD shows better scalability as compared with most other methods which shows that the growth of distribution function time is lower for NMEEF-SD than the classical beam search and the exhaustive search based method. APRIORI-SD exhibits lower scalability as compared with all the other methods in terms of both execution time and memory usage. It is because of the fact that it employs the exhaustive search strategy for generating candidate subgroups. Although SD-Map implements an exhaustive search strategy, it

Table 4. Execution Time (s)

	SD	CN2-SD	APRIORI-SD	SD-MAP	DSSD	NMEEF-SD
Adult	1 122	2 064	No	12	8	10
Bcw	10	26	8	8	3	4
Census-income	-	-	-	180	-	-
Diabetes	5	11	9	1	0	1
Germand	20	43	No	3	1	2
Heart	4	10	165	1	0	1
Hypothyroid	38	229	No	4	1	2
Iris	3	8	89	0	0	1
Kr-vs-kp	42	312	No	5	2	3
Mushroom	57	882	No	3	0	3
Sick	32	263	No	6	1	3
Tic-Tac-Toe	7	13	10	3	0	2
Vote	6	11	192	2	1	2

Table 5. Peak Memory Usage (MB)

	SD	CN2-SD	APRIORI-SD	SD-MAP	DSSD	NMEEF-SD
AdultAll	964.8	192.4	No	1 014.8	188.34	190.85
Bcw	31.4	23.9	34.8	212.2	22.37	23.90
Census-income	-	-	-	5 043.4	-	-
Diabetes	15.3	12.8	17.7	16.4	10.24	11.73
Germand	22.2	25.0	No	533.3	18.85	20.06
Heart	12.4	15.9	38.4	44.8	9.70	10.97
Hypothyroid	42.7	221.6	No	824.6	29.42	35.47
Iris	12.2	13.5	26.8	18.8	11.05	12.78
Kr-vs-kp	51.5	341.9	No	917.8	32.26	45.60
Mushroom	102.9	513.2	No	928.5	8.24	10.70
Sick	33.6	272.3	No	313.7	25.03	28.23
Tic-Tac-Toe	18.8	13.7	21.9	22.3	11.15	14.75
Vote	13.3	17.9	44.1	56.6	11.94	12.29

shows better performance as compared with APRIORI-SD since it uses a divide-and-conquer strategy while APRIORI-SD uses the candidate-generation-and-test strategy. This feature leads to the generation of exponential number of candidates with the increase in data size.

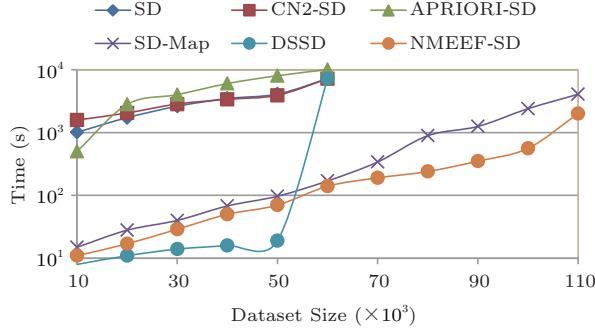


Fig.2. Scalability in terms of execution time.

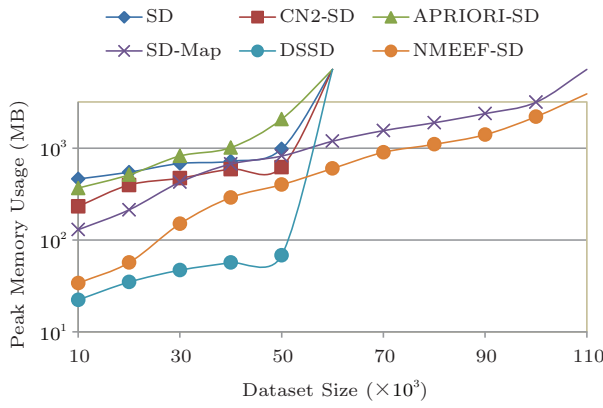


Fig.3. Scalability in terms of peak memory usage.

5.4.3 Evaluation

The quality of subgroups generated by each algorithm can be viewed from Table 6. The result can be summarized as follows.

1) The subgroups generated by DSSD method have the highest average coverage and support.

2) DSSD generated subgroups have the highest average length.

3) DSSD induces better subgroups in terms of significance which is computed in terms of the average likelihood ratio of a subgroup.

4) Subgroups discovered by DSSD have the highest quality (according to the unusualness measure).

5) APRIORI-SD generated subgroups have the lowest redundancy and have the highest overall coverage both within the class and the datasets.

The interpretation of the results is listed as follows.

1) In the first iteration of a standard beam search, a predefined number (beam width, bw) of subgroups with depth 1 are generated. In the next iteration, candidate subgroups with depth 2 are generated from the previously generated depth 1 subgroups and bw subgroups are selected according to the quality. Thus candidate subgroups with depth n are generated from the subgroups of depth $n - 1$ in each level of beam search, and among them, only bw subgroups are selected as quality basis. Hence, the incorporation of the beam width parameter imposes a restriction on the number of generated subgroups; the discovered subgroup set suffers from a loss of quality subgroups. The level of exploration becomes limited in this type of search. On the contrary, DSSD at first mines a large number of j subgroups (which include most of the subgroups with different depths according to quality) and there it selects top- k . Moreover, this method employs both “=” and “!=” parameters for subgroup refinement which allows to cover on average a large number of examples within the class and within the dataset.

2) For DSSD, the maximum depth of subgroup is fixed as 5 which leads this method to have the longest average subgroup length. Despite this character, DSSD

Table 6. Comparative Performance

Method	Coverage		Support		Number of Subgroups	Length of Subgroup	Significance (Average)	WRAcc		Confidence (Average)	Redundancy* (%)
	Average	Overall	Average	Overall				Average	Maximum		
SD	0.069 6	0.182 9	0.049 3	0.132 0	100	3.252	37.253 0	0.030 0	0.132 0	0.796 5	44.09
CN2-SD	0.078 1	0.285 9	0.068 7	0.101 0	100	3.248	138.743 0	0.044 9	0.162 3	0.802 3	76.79
APRIORI-SD	0.169 2	0.479 5	0.162 2	0.355 6	105	3.337	13.347 0	0.059 3	0.176 9	0.863 7	9.32
SD-Map	0.206 8	0.272 6	0.179 3	0.213 8	100	3.271	622.659 4	0.072 1	0.103 7	0.731 4	87.38
DSSD	0.318 9	0.422 2	0.229 8	0.288 5	100	4.285	778.583 0	0.114 0	0.118 0	0.622 1	81.17
NMEEF-SD	0.282 3	0.451 2	0.182 9	0.344 6	95	4.060	711.560 0	0.107 0	0.172 0	0.854 6	39.96

Note: *: for evaluating the subgroups generated by DSSD, each of the subgroups is treated as individual rather than a set for the ease of comparison.

has the highest average coverage. This is due to the fact that DSSD refines subgroups by employing both “=” and “!=” operators while other subgroup discovery methods (only those used for experiments) only use “=” operator for generating subgroups. This feature imposes multiple conditions on the same attribute while removing a number of uninteresting tuples at the same time; thus it allows to cover more examples as compared with other methods.

3) The significance criterion evaluates a subgroup by measuring its distributional unusualness unbiased to any particular target class. The generality of DSSD algorithm increases in the first two phases. In the first phase, the subgroup selection strategy removes a subgroup if it has the same quality as any of its generalization. In the second phase, a heuristic known as dominance-based pruning is applied on the selected subgroups of the previous phase. This heuristic considers each of the conditions of a subgroup description one by one and removes them if it does not decrease the quality of the subgroup. The resulting subgroup set is more general and hence covers a number of both target and non-target class examples. Therefore DSSD generated subgroups are highly significant.

4) DSSD post-processes the initially generated j subgroups by implementing dominance-based pruning which prunes the condition of a subgroup description if it does not decrease the subgroup's quality. This heuristic allows DSSD to have its subgroup description simpler while increasing its average quality. Moreover, the use of “!=” operator allows to cover a large number of examples by a subgroup which essentially increases the subgroup's quality.

5) In APRIORI-SD, a restriction is imposed on similar subgroup generation by employing a new parameter representing how many times an example can be covered by a subgroup. Moreover, this algorithm implements the weighted WRAcc heuristic which allows decreasing the weight of an example when it is covered by a subgroup and considering the new weight for measuring the WRAcc for evaluating next subgroup. These features allow this method to generate less redundant subgroups by decreasing the chance to cover overlapping examples. As a result, the generated subgroup sets cover a large number of examples within the dataset. In APRIORI-SD, candidate subgroups are initially selected according to the minimum confidence threshold. The confidence measures the fraction of a subgroup that corresponds to a particular target class. From the result table, it is observed that the subgroups discovered

by APRIORI-SD have the highest average confidence which clearly states that these subgroups are highly specific to the target class. This is also apparent from the amount of target coverage which is the highest for this method.

Consider the following two graphs representing the coverage and support of different methods used in the experiments. These graphs depict rule coverage and support by different SD methods which show that APRIORI-SD has the highest difference between its average and the overall coverage and support. This finding is very obvious as this method has less redundancy rate, and hence it has less number of overlapping examples. In contrast, DSSD has the lowest difference between its overall and average coverage and support because of generating a high number of overlapping examples.

5.4.4 Comparison and Discussion

It is observed from the experiments that NMEEF-SD is more efficient than the traditional beam search based methods (SD and CN2-SD) and the exhaustive search methods (APRIORI-SD and SD-Map) in terms of execution time. This is due to the fact that the time complexity of NMEEF-SD is proportional to the number of classes and instances of a dataset while for the other methods, the complexity is related to the number of variables and instances.

The experiments also follow that the exhaustive search based methods have poorer scalability as compared with the traditional beam search based methods and evolutionary method. As for the exhaustive methods, a number of generated candidates grow exponentially with the increase in the size of instances.

DSSD generated subgroups exhibit the highest average support; still they have the lowest confidence. It is because of the fact that DSSD generated subgroups are more general. The average subgroup size is around 2971 among which only 1626 are covered by the target class. Though DSSD generated j subgroups are specific at the first phase, their generality increases by implementing dominance pruning in the second phase of the algorithm.

DSSD generates high quality subgroups, which is evident from the fact that the average quality and the maximum quality of subgroups (generated by DSSD) are nearly equal. In the traditional beam search, a maximum of bw subgroups can be generated and finally bw subgroups should be returned according to the quality. For example, consider beam width is 20. In

the first iteration of beam search, 20 subgroups will be generated with depth 1. In the second iteration, another 20 subgroups will be generated of depth 2 from the subgroups of the first iteration. It is quite possible that if there was no restriction on the number of generated solutions, in the first iteration, this method would generate higher quality subgroups as compared with those generated from the latter iterations.

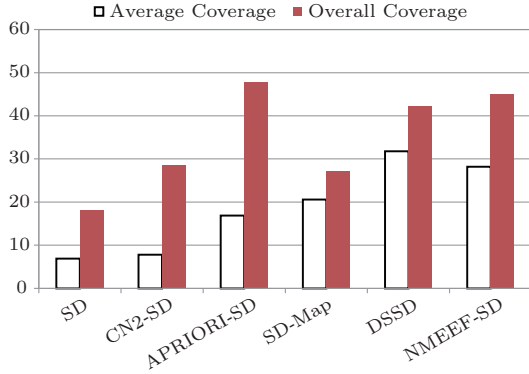


Fig.4. Coverage of different SD methods.

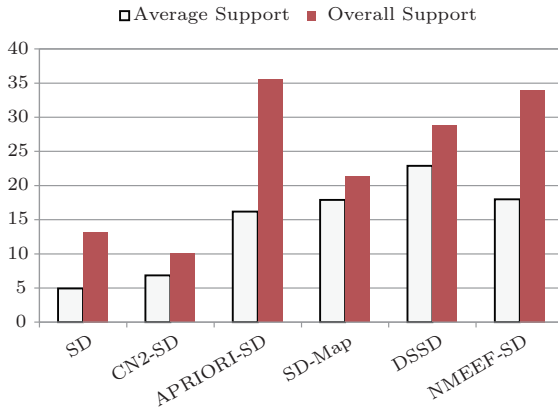


Fig.5. Support of different SD methods.

Although CN2-SD implements beam search for generating candidates, it generates higher quality subgroups as compared with SD. This algorithm uses a modified unusualness for extracting subgroups. This strategy allows to decrease the weight of an example when it is covered by any subgroup. The modified weight of all the examples is taken into consideration for measuring the unusualness of a subgroup.

APRIORI-SD generated subgroups are of higher quality than those of CN2-SD. APRIORI-SD extracts the subgroups having a minimum support and confidence value in the initial phase of the algorithm, and then post-processes the subgroups according to the

unusualness measure; whereas CN2-SD extracts subgroups using the unusualness measure at the very initial stage and loses some quality subgroups.

NMEEF-SD generated subgroups are of higher quality than the generated subgroups of the standard beam search based methods and the exhaustive search based methods. This method does not use any pruning strategy but lets the rule evolve, which has a minimum fitness score.

As exhaustive search generates every possible candidate, it is better to use this strategy when the problem size is smaller or when a problem-specific heuristic can be employed to reduce the number of candidates. The cost of this type of search is proportional to the number of generated candidates; hence beam search and genetic algorithm based search are the alternative heuristics for searching candidates when the search space is very large. Different methods implement different measures for ranking subgroups. The interestingness of a subgroup depends on its unusualness and size. That is why the quality measure needs to combine both factors. The weighted relative accuracy $WRAcc$ takes these two factors into consideration when ranking subgroups.

Different algorithms employ different strategies for reducing the number of redundant subgroups. The algorithms, CN2-SD, APRIORI-SD and RSD, implement the weighted covering scheme for restricting the number of redundant subgroups. Although this strategy imposes a restriction on a number of redundant subgroups, it is still not able to remove them completely. Another non-redundant subgroup discovery algorithm DSSD implements three non-redundant subgroup selection strategies. According to these approaches, if a subgroup has its generalization with the same quality in the discovered subgroup set, it should be removed.

This may not be always feasible as in some application domain the specific subgroups (as compared with its generalization) may be preferred. For example, in medical domain, suppose symptoms A and B diagnose disease D , and again symptoms A , B and C diagnose the same disease. Though both of the symptom groups detect the same disease D , the severity of the disease D may vary for each of the symptom groups, i.e., patients with symptoms A , B and C may have disease D with higher severity. Therefore whether to choose the most general or specific subgroups solely depends on the application domain, and the method should be chosen accordingly.

Incorporating domain knowledge in subgroup discovery methods may be useful for extracting quality

subgroups with respect to a particular application area. Using domain knowledge for discovering subgroups will also be helpful for restricting the search space.

A number of challenges exist in the domain of subgroup discovery. Major challenge lies on the performance of each of the methods. From the result, we have already seen that a method may have higher preciseness, i.e., lower redundancy, but may not be able to provide a high quality subgroup. This fact is depicted from the performance of APRIORI-SD algorithm. Similarly, a method may have higher generality and quality but may generate the rule of lower confidence. However, it is very vital to learn which features of an algorithm should be considered for generating quality subgroups for a specific application domain.

6 Conclusions

Different subgroup discovery approaches use different pruning techniques and measures for selecting and ranking candidate subgroups. An efficient pruning technique should not eliminate any significant candidates. At the same time, a good measure should consider both the unusualness and the size of the subgroups for evaluation.

In this paper, we compared several subgroup discovery approaches in terms of their efficiency, scalability (with respect to execution time and peak memory usage) and also evaluated the discovered subgroups by employing several measures. This work has proposed three new measures as it is not sufficient to evaluate subgroups in the light of the existing measures. Furthermore, the evaluation and comparison has been performed by incorporating 13 different datasets from the UCI machine learning repository which are diverse both in the size of instances and the attributes. The interpretation of the comparison allows to provide a detailed understanding of different methods, to find out the gaps of the existing methods and to give some suggestions about the features that should be considered for domain specific subgroup discovery.

The experiments showed that the exhaustive search based methods exhibit better performance compared with other methods when the problem size is smaller. Moreover, it is also evident from the experiments that methods that employ the unusualness as the quality measure perform well compared with other algorithms as this measure considers both the unusualness and the size of the subgroups. The resulting analysis will be helpful in the future for developing an efficient subgroup discovery method.

References

- [1] Fayyad U, Piatetsky-Shapiro G, Smyth P. Knowledge discovery and data mining: Towards a unifying framework. In *Proc. the 2nd International Conference on Knowledge Discovery and Data Mining (KDD)*, Aug. 1996, pp.82-88.
- [2] Novak P K, Lavrač N, Webb G I. Supervised descriptive rule discovery: A unifying survey of contrast set, emerging pattern and subgroup mining. *The Journal of Machine Learning Research*, 2009, 10: 377-403.
- [3] Gamberger D, Lavrač N, Krstačić G. Active subgroup mining: A case study in coronary heart disease risk group detection. *Artificial Intelligence in Medicine*, 2003, 28(1): 27-57.
- [4] Gamberger D, Lavrač N. Supporting factors in descriptive analysis of brain ischaemia. In *Proc. the 11th Conference on Artificial Intelligence in Medicine (AIME)*, Jul. 2007, pp.155-159.
- [5] Gamberger D, Lavrač N, Krstačić A, Krstačić G. Clinical data analysis based on iterative subgroup discovery: Experiments in brain ischaemia data analysis. *Applied Intelligence*, 2007, 27(3): 205-217.
- [6] Klösigen W. Applications and research problems of subgroup mining. In *Proc. the 11th ISMIS*, June 1999.
- [7] Lavrač N, Cestnik B, Gamberger D, Flach P. Decision support through subgroup discovery: Three case studies and the lessons learned. *Machine Learning*, 2004, 57(1/2): 115-143.
- [8] Romero C, González P, Ventura S, del Jesus M J, Herrera F. Evolutionary algorithms for subgroup discovery in e-learning: A practical application using Moodle data. *Expert Systems with Applications: An International Journal*, 2009, 36(2): 1632-1644.
- [9] Klösigen W, May M. Spatial subgroup mining integrated in an object-relational spatial database. In *Proc. the 6th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD)*, Aug. 2002, pp.275-286.
- [10] May M, Ragia L. Spatial subgroup discovery applied to the analysis of vegetation data. In *Proc. the 4th Practical Aspects of Knowledge Management*, Dec. 2002, pp.49-61.
- [11] Gamberger D, Lavrač N. Expert-guided subgroup discovery: Methodology and application. *Journal of Artificial Intelligence Research*, 2002, 17(1): 501-527.
- [12] Kavšek B, Lavrač N, Jovanoski U. APRIORI-SD: Adapting association rule learning to subgroup discovery. In *Proc. the 5th IDA*, Aug. 2003, pp.230-241.
- [13] Atzmueller M, Puppe F. SD-Map — A fast algorithm for exhaustive subgroup discovery. In *Proc. the 10th European Conference on Principle and Practice of Knowledge Discovery in Databases (PKDD)*, Sept. 2006, pp.6-17.
- [14] Leeuwen M, Knobbe A. Diverse subgroup set discovery. *Data Mining and Knowledge Discovery*, 2012, 25(2): 208-242.
- [15] del Jesus M J, González P, Herrera F, Mesonero M. Evolutionary fuzzy rule induction process for subgroup discovery: A case study in marketing. *IEEE Trans. Fuzzy Systems*, 2007, 15(4): 578-592.

- [16] Herrera F, Carmona C J, González P, del Jesus M J. An overview on subgroup discovery: Foundations and applications. *Knowledge Information System*, 2011, 29(3): 495-525.
- [17] Klösgen W. Explora: A multipattern and multistrategy discovery assistant. In *Advances in Knowledge Discovery and Data Mining*, Fayyad V M, Piatetsky-Shapiro G, Smyth P et al. (eds.), AAAI/WIT Press, 1996, pp.249-271.
- [18] Wrobel S. An algorithm for multi-relational discovery of subgroups. In *Proc. the 1st European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD)*, Jun. 1997, pp.78-87.
- [19] Han J, Cheng H, Xin D, Yan X. Frequent pattern mining: Current status and future directions. *Data Mining and Knowledge Discovery*, 2007, 15(1): 55-86.
- [20] Grosskreutz H, Rüping S, Wrobel S. Tight optimistic estimates for fast subgroup discovery. In *Proc. the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, Sept. 2008, pp.440-456.
- [21] Boley M, Grosskreutz H. Non-redundant subgroup discovery using a closure system. In *Proc. the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, Sept. 2009, pp.179-194.
- [22] Grosskreutz H, Rüping S. On subgroup discovery in numerical domains. *Data Mining and Knowledge Discovery*, 2009, 19(2): 210-226.
- [23] Lavrač N, Kavšek B, Flach P, Todorovski L. Subgroup discovery with CN2-SD. *The Journal of Machine Learning Research*, 2004, 5: 153-188.
- [24] Atzmueller M, Puppe F, Buscher H P. Towards knowledge-intensive subgroup discovery. In *Proc. the Lernen-Wissensentdeckung-Adaptivität-Fachgruppe Maschinelles Lernen*, Oct. 2004, pp.111-117.
- [25] Klösgen W, May M, Petch J. Mining census data for spatial effects on mortality. *Intelligent Data Analysis*, 2003, 7(6): 521-540.
- [25] Clark P, Niblett T. The CN2 induction algorithm. *Journal of Machine Learning*, 1989, 3(4): 261-283.
- [26] Lavrač N, Železný F, Flach P. RSD: Relational subgroup discovery through first-order feature construction. In *Proc. the 12th International Conference on Inductive Logic Programming*, Jul. 2002, pp.149-165.
- [27] Jovanoski V, Lavrač N. Classification rule learning with APRIORI-C. In *Proc. the 10th Portuguese Conference on Artificial Intelligence*, Dec. 2001, pp.44-51.
- [28] Han J, Pei J, Yin Y. Mining frequent patterns without candidate generation. In *Proc. the ACM SIGMOD International Conference on Management of Data*, May 2000, pp.1-12.
- [29] Agrawal R, Srikant R. Fast algorithms for mining association. In *Proc. the 20th VLDB*, Sept. 1994, pp.487-499.
- [30] del Jesus M J, González P, Herrera F. Multiobjective genetic algorithm for extracting subgroup discovery fuzzy rules. In *Proc. IEEE Symp. Computational Intelligence in Multicriteria Decision Making*, Apr. 2007, pp.50-57.
- [31] Zitzler E, Laumanns M, Thiele L. SPEA2: Improving the strength Pareto evolutionary algorithm. In *Proc. International Congress on Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, Sept. 2001, pp.95-100.
- [32] Carmona C J, González P, del Jesus M J, Herrera F. NMEEF-SD: Non-dominated multiobjective evolutionary algorithm for extracting fuzzy rules in subgroup discovery. *IEEE Trans. Fuzzy Systems*, 2010, 18(5): 958-970.
- [33] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm NSGA-II. *IEEE Trans. Evolutionary Computation*, 2002, 6(2): 182-197.
- [34] Luna J M, Romero J R, Romero C, Ventura S. On the use of genetic programming for mining comprehensible rules in subgroup discovery. *IEEE Trans. Cybernetics*, 2014, 44(12): 2329-2341.
- [35] Gamberger D, Lavrač N. Generating actionable knowledge by expert-guided subgroup discovery. In *Proc. the 6th European Conference on Principles of Data Mining and Knowledge Discovery*, Aug. 2002, pp.163-175.
- [36] Lavrač N. Subgroup discovery techniques and applications. In *Proc. the 9th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, May 2005, pp.2-14.
- [37] Carmona C J, González P, del Jesus M J, Navío-Acosta M, Jiménez-Trevino L. Evolutionary fuzzy rule extraction for subgroup discovery in a psychiatric emergency department. *Soft Computing*, 2011, 15(12): 2435-2448.
- [38] Carmona C J, Ruiz-Rodado V, del Jesus M J, Weber A, Grootveld M, González P, Elizondo D. A fuzzy genetic programming-based algorithm for subgroup discovery and the application to one problem of pathogenesis of acute sore throat conditions in humans. *Information Sciences*, 2015, 298(C): 180-197.
- [39] Gamberger D, Lavrač N. Avoiding data overfitting in scientific discovery: Experiments in functional genomics. In *Proc. the 16th European Conference on Artificial Intelligence*, Aug. 2004, pp.470-474.
- [40] Mueller M, Rosales R, Steck H, Krishnan S, Rao B, Kramer S. Subgroup discovery for test selection: A novel approach and its application to breast cancer diagnosis. In *Proc. the 8th Intelligent Data Analysis*, Aug.31-Sept.2, 2009, pp.119-130.
- [41] Trajkovski I, Železný F, Lavrač N, Tolar J. Learning relational descriptions of differentially expressed gene groups. *IEEE Trans. Systems, Man, and Cybernetics*, 2008, 38(1): 16-25.
- [42] Trajkovski I, Železný F, Tolar J, Lavrač N. Relational subgroup discovery for descriptive analysis of microarray data. In *Proc. the 2nd International Conference on Computational Life Sciences*, Sept. 2006, pp.86-96.
- [43] Schmidt J, Hapfelmeier A, Mueller M, Perneczky R, Kurz A, Drzezga A, Kramer S. Interpreting PET scans by structured patient data: A data mining case study in dementia research. *Knowledge and Information Systems*, 2010, 24(1): 149-170.

- [45] Kavšek B, Lavrač N. Using subgroup discovery to analyze the UK traffic data. *Advances in Methodology and Statistics*, 2004, 1(1): 249-264.
- [46] Kavšek B, Lavrač N, Bullas J C. Rule induction for subgroup discovery: A case study in mining UK traffic accident data. In *Proc. International Multi-Conference on Information Society*, Jan. 2002, pp.127-130.
- [47] Agrawal R, Mannila H, Srikant R, Toivonen H, Verkamo A I. Fast discovery of association rules. In *Advances in Knowledge Discovery and Data Mining*, Fayyad VM, Piatetsky-Shapiro G, Smyth P *et al.* (eds.), AAAI/MIT Press, 1996, pp.307-328.
- [48] Lavrač N, Flach P, Zupan B. Rule evaluation measures: A unifying view. In *Proc. the 9th International Workshop on Inductive Logic Programming (ILP)*, Jun. 1999, pp.174-185.
- [49] Lichman M. UCI machine learning repository, 2013. <http://archive.ics.uci.edu/ml>, Mar. 2016.
- [50] Kohavi R, Sommerfield D, Dougherty J. Data mining using MLC++: A machine learning library in C++. *International Journal on Artificial Intelligence Tools*, 1997, 6(4): 537-566.
- [51] Demšar J, Curk T, Erjavec A, Gorup C, Hočevár T, Milutinović M, Možina M, Polajnar M, Toplak M, Starič A, Štajdohar M, Umek L, Žagar L, Žbontar J, Žitnik M, Zupan B. Orange: Data mining toolbox in python. *Journal of Machine Learning Research*, 2013, 14: 2349-2353.
- [52] Atzmueller M, Lemmerich F. VIKAMINE — Open-source subgroup discovery, pattern mining, and analytics. In *Proc. ECML PKDD*, Sept. 2012, pp.842-845.
- [53] Alcalá-Fdez J, Sánchez L, García S, del Jesus M J, Ventura S, Garrell J M, Otero J, Romero C, Bacardit J, Rivas V M, Fernández J C, Herrera F. KEEL: A software tool to assess evolutionary algorithms for data mining problems. *Soft Computing*, 2009, 13(3): 307-318.



Sumyea Helal received her B.Sc. degree in computer science and engineering from the Rajshahi University of Engineering and Technology, Bangladesh, in 2005, and M.Sc. degree in information technology from the Royal Institute of Technology, Sweden, in 2010. Right now she is pursuing her Ph.D. degree in computer science at the University of South Australia, Adelaide. From 2006 to 2011, Sumyea worked as a lecturer in Computer Science and Engineering Department at Ahsanullah University of Science and Technology, Bangladesh. Her research interests lie in data mining especially in the field of pattern and knowledge discovery.