Local search can improve fixed-size redescription set construction

Matej Mihelčić[1,2][0000−0002−1023−8413]

Faculty of Science, University of Zagreb, Zagreb, Croatia matmih@math.hr
Ruđer Bošković Institute, Zagreb, Croatia matej.mihelcic@irb.hr

Abstract. Redescription mining is a broadly applicable field of data mining tasked with creating algorithms that discover subsets of data that allow multiple rule-based characterizations. Discovering re-descriptions is important in many scientific fields such as biology, medicine and ecology. However, as with many other data mining algorithms, the result of redescription mining can be a large set of objects, making it hard to analyze and understand by the domain experts. Thus, recent efforts aim to provide smaller sets capturing the essence of information that is to be conveyed to the experts. One such approach is to produce a redescription set of a fixed size, where its cardinality is set by the user. The main technique to produce fixed size sets is to use a multi-objective optimisation with scalarisation. The result is a greedy heuristic approach that iteratively selects a Pareto optimal redescription from a large pool of redescriptions and adds it to the set being constructed. In this work, we show that such output sets can be improved using local search procedure, opening the potential for future research of various heuristic approaches for fixed sized redescription set construction.

Keywords: Redescription set construction · Local search · Multi-objective optimization.

1 Introduction

Various data mining tasks, such as itemset mining [5], association rule mining [1], subgroup discovery [15, 24] and redescription mining [21] are tasked with development of algorithmic approaches that allow detection and interpretable presentation of various types of knowledge contained in the input dataset. This knowledge is often presented as a set of rule-based objects, where each object describes a different subset of data instances. Given a set of n instances and the minimal size of a subset ms, one can produce $\sum_{k=ms}^{n} \binom{n}{k}$ different k-subsets of a set of n instances. This demonstrates that the size of data mining algorithm outputs can potentially be very large. Additionally, multiple rule-based objects can describe the same subset of instances, making this problem even more pronounced.

Because of this, the majority of recent data mining algorithms contain techniques to remove insignificant patterns, duplicates or very similar objects. These

techniques include traversing the lattice of itemsets [1], using beam search [17], greedy updates of fixed size with various filters [7], alternations with trees of fixed depth with various filters [21, 19] etc. Despite these algorithmic advancements, the size of their outputs can still be too large to effectively manage. Thus, approaches from various fields define optimization functions that capture the essence of the properties of what should be the representable or desired output set for a given task and the application domain (e.g. [16, 19]). Then, often by performing a combinatorial multi-objective optimization, they select only a subset of produced objects that maximize or minimize the objective function.

In this work, we study a multi-objective minimization problem designed to create a fixed-size redescription sets [19], allowing user-defined preference to be incorporated into the optimization process. This task differs from a standard multi-objective optimization problem where one searches for the whole Pareto front, since many non-dominated solutions are not usable in the studied domain. We empirically compare the performance of the existing greedy approach [19] that constructs a fixed size redescription set from a larger pool of available redescriptions to the output sets produced by adopting local search procedure for this task, using identical pool of redescriptions. Additionally, we compute the lower bound on quality scores of the output sets given a predefined cardinality of the output sets and the available pool of redescriptions for five studied problems consisting of two-views each.

2 Definitions and notation

The input datasets for a task of redescription mining consist of a sequence of data tables. The set of all attributes is denoted V. All data tables contain the same set of instances E, but each data table contains a mutually disjoint set of attributes, which is a subset of V. A set of attribute group mappings, called views, are denoted $\mathcal{W} = \{W_1, W_2, \ldots, W_k\}$. For an instance $e_i \in E$, $W_p(e_i, j)$ returns the value of the j-th attribute in the p-th view. In general, views can contain mutually different numbers of attributes. A query q is a logical formula consisting of a set of numerical, categorical or Boolean attributes with corresponding attribute value constraints and logical connectives: conjunction, disjunction or negation. The set of all instances described by a query is denoted $supp(q)$. A redescription $R = (q_1, q_2, \ldots, q_k)$ is a tuple of queries, where q_j contains attributes from W_j. $supp(R) = supp(q_1) \cap supp(q_2) \cap \cdots \cap supp(q_k)$ denotes the support set of a redescription. $attr(R)$ and $attrs(R)$ denote the multi-set and set of all attributes contained in the queries of R. $o = |supp(R)|$ is a redescription support set size and $p_i = |supp(q_i)|/|E|$ a query marginal probability. We denote a set of redescriptions with $\mathcal{R}$ and $m = |\mathcal{R}_{output}|$. Important redescription and redescription set evaluation measures can be seen in Table 1.

Definition 1. *Given a set of instances E, a set of attributes V describing these instances, a set of views $\mathcal{W}$, a query language Q, a similarity measure $\mathcal{M}$ and a constraint set $\mathcal{C}$, the task of redescription mining is to discover all redescriptions satisfying constraints in $\mathcal{C}$.*

Table 1. Redescription and redescription set evaluation measures.

Measure definition	Description
$J(R) = \frac{\lvert supp(q_1) \cap supp(q_2) \cap \cdots \cap supp(q_k) \rvert}{\lvert supp(q_1) \cup supp(q_2) \cup \cdots \cup supp(q_k) \rvert}$	Redescription accuracy
$p(R) = \sum_{n=o}^{\lvert E \rvert} \binom{\lvert E \rvert}{n} (\prod_{i=1}^{k} p_i)^n \cdot (1 - \prod_{i=1}^{k} p_i)^{\lvert E \rvert - n}$	Redescription statistical significance
$AEJ(R_i) = \frac{1}{\lvert \mathcal{R} \rvert - 1} \cdot \sum_{j=1}^{\lvert \mathcal{R} \rvert} J(supp(R_i), supp(R_j)),\ i \neq j$	Average redescription instance redundancy
$AAJ(R_i) = \frac{1}{\lvert \mathcal{R} \rvert - 1} \cdot \sum_{j=1}^{\lvert \mathcal{R} \rvert} J(attrs(R_i), attrs(R_j)),\ i \neq j$	Average redescription attribute redundancy
$comp(R) = \lvert attrs(R) \rvert / c,\ \lvert attrs(R) \rvert < c$ $comp(R) = 1, c \leq \lvert attrs(R) \rvert$	Redescription complexity c - user defined
$p_{sc}(R) = log_{10}(p(R))/17 + 1,\ p(R) \geq 10^{-17}$ $p_{sc}(R) = 0,\ p(R) < 10^{-17}$	Redescription significance score
$AEJS(\mathcal{R}) = \sum_{R \in \mathcal{R}} AEJ(R) / \lvert \mathcal{R} \rvert$	Redescription set instance redundancy
$AAJS(\mathcal{R}) = \sum_{R \in \mathcal{R}} AAJ(R) / \lvert \mathcal{R} \rvert$	Redescription set attribute redundancy
$AJS(\mathcal{R}) = \sum_{R \in \mathcal{R}} (1 - J(R)) / \lvert \mathcal{R} \rvert$	Redescription set accuracy score
$ApS(\mathcal{R}) = \sum_{R \in \mathcal{R}} (p_{sc}(R)) / \lvert \mathcal{R} \rvert$	Redescription set statistical significance score
$AcompS(\mathcal{R}) = \sum_{R \in \mathcal{R}} (comp(R)) / \lvert \mathcal{R} \rvert$	Redescription set complexity score
$ERCS(\mathcal{R}) = 1 - \lvert \cup_{R \in \mathcal{R}} supp(R) \rvert / \lvert E \rvert$	Redescription set coverage of instances
$ARCS(\mathcal{R}) = 1 - \lvert \cup_{R \in \mathcal{R}} attrs(R) \rvert / \lvert V \rvert$	Redescription set coverage of attributes
$w_i \geq 0$	User-defined preference weights
$min_{\mathcal{R} \subseteq \mathcal{R}_{all},\ \lvert \mathcal{R} \rvert = m}\ w_1 \cdot AJS(\mathcal{R}) + w_2 \cdot ApS(\mathcal{R}) +$ $w_3 \cdot AEJS(\mathcal{R}) + w_4 \cdot AAJS(\mathcal{R}) + w_5 \cdot AcompS(\mathcal{R}) +$ $w_6 \cdot ERCS(\mathcal{R}) + w_7 \cdot ARCS(\mathcal{R})$	Redescription set optimization function

$\mathcal{M}$ equals the Jaccard index [11] and $\mathcal{C}$ usually contains minimal redescription accuracy, maximal allowed redescription p-value and minimal/maximal redescription support set size. A query language Q defines a set of valid queries.

3 Related work

Mihelčić et al. [19] proposed a procedure to generate redescription sets of a fixed size, taking into account user-defined preferences. A specifically designed multi-objective function, used by the procedure, incorporates various redescription and redescription set quality criteria. The resulting multi-objective optimization problem is tackled using a scalarization method [9] in a combinatorial setting. The solution to this set function minimization problem is a fixed-sized subset of a set of all available redescriptions, where the output redescription set size is defined by the user. Kalofolias et al. [13] presented an approach for redescription ranking. It is related but strictly different task from creating a fixed-size representative redescription set from potentially large sets of created redescriptions. Lakkaraju et al. [16] defined a submodular multi-objective function to evaluate decision sets. The function is formulated so that the maximum of the function describes the best subset of rules (according to the evaluation criteria) from a set of available rules. The resulting submodular set function maximization problem was solved using the heuristic smooth local search [4].

Local search was used to create decision rules [3], to search for partial classification rules [22], in learning locally interpretable rule ensembles [14] and for rule induction [10]. It was also used for multi-objective balanced classification [23] and mining Pittsburgh classification rules [12].

4 Preliminaries on a greedy approach for fixed-size redescription set construction

The greedy procedure [19] iteratively adds redescriptions from the set of all produced redescriptions $\mathcal{R}_{all}$ into a redescription set $\mathcal{R}_{out}$ so that:

a) the first redescription is selected by computing:

$$R_{first} = argmin_{R\in\mathcal{R}_{all}}(w_1 \cdot (1 - J(R)) + w_2 \cdot p_{sc}(R) + w_3 \cdot score_{ocurEl}(R) + w_4 \cdot score_{ocurAt}(R) + w_5 \cdot comp(R)) \quad (1)$$

, where $score_{ocurEl}$ measures the occurrence of instances in redescription support sets, and $score_{ocurAt}$ the occurrence of attributes in redescription queries of redescriptions contained in $\mathcal{R}_{all}$. A redescription describing infrequently occuring instances with infrequently occurring attributes in $\mathcal{R}_{all}$ is used as the initial (seed) redescription in redescription set construction.

b) other redescriptions are selected by computing:

$$R_{best} = argmin_{R\in\mathcal{R}_{all}} w_1 \cdot (1 - J(R)) + w_2 \cdot (\frac{k}{m} p_{sc}(R) + (1 - \frac{k}{m}) \cdot \frac{|supp(R)|}{|E|}) + w_3 \cdot score_{elemSim,\mathcal{R}_{out}}(R) + w_4 \cdot score_{attrSim,\mathcal{R}_{out}}(R) + w_5 \cdot comp(R) \quad (2)$$

, where k denotes the size of a redescription set under construction in the current iteration, $score_{elemSim,\mathcal{R}_{out}}(R)$ is the maximum Jaccard index between $supp(R)$ and any redescription contained in the $\mathcal{R}_{out}$ and $score_{attrSim,\mathcal{R}_{out}}(R)$ is the maximal Jaccard index between $attrs(R)$ and a set of attributes of any redescription contained in the redescription set under construction $\mathcal{R}_{out}$.

The procedure does not explicitly incorporate coverage, but it regulates the weight of support size importance during redescription set construction. This, in combination with redundancy measures, increases attribute/instance coverage of the $\mathcal{R}_{out}$. It allows faster computation over the approach explicitly computing coverage. There are two potential sources of error that can be improved with the local search procedure:

1. Errors introduced due to the choice of the seed redescription.
2. Errors introduced by the greedy selection of candidate redescriptions. The RSCP, with $w_i > 0, \forall i$, at each step chooses the Pareto optimal solution, given current R_{out}. However, dependencies between redescriptions make it necessary to occasionally choose a redescription with a sub-optimal score value to obtain the best possible $\mathcal{R}_{out}$.

5 The local search procedure for fixed-size redescription set construction

The local search procedure first constructs an initial set using steps as in Greedy approach from [19], but using a slightly modified optimisation function. Next, it creates m candidate sets, where each set is constructed from a seed redescription corresponding to one of the m redescriptions from the initial set of redescriptions. This step aims to alleviate the first source of error of the greedy procedure. Next, these sets are iteratively improved using a local search procedure by removing a predefined number of redescriptions from the candidate sets and then completing these sets using a greedy procedure. By removing random subsets of various sizes from candidate sets, we address the second source of errors introduced by the greedy procedure.

The local search procedure requires a methodology to complete redescription sets from which a number of randomly selected redescriptions were removed. Procedure from equation 2 is unsuitable since preference weight of redescription support size changes with the size of the redescription set under construction, irrespective of redescriptions already contained in it. Such procedure is adequate for constructing redescription sets anew but not for their completion. Newly developed selection procedure from equations 3 overcomes the discussed drawback by incorporating entity coverage criteria as a weight instead of current output redescription set size.

$$R_{best} = argmin_{R \in \mathcal{R}_{all}} w_1 \cdot (1 - J(R)) + w_2 \cdot (1 - ERCS(\mathcal{R}_{out}) p_{sc}(R) + (ERCS(\mathcal{R}_{out})) \cdot \frac{|supp(R)|}{|E|}) + w_3 \cdot score_{elemSim,\mathcal{R}_{out}}(R) + w_4 \cdot score_{attrSim,\mathcal{R}_{out}}(R)) + w_5 \cdot comp(R) + w_6 \cdot ERCS_{new}(R) + w_7 \cdot ARCS_{new}(R) \quad (3)$$

$ERCS_{new}(R) = 1 - \frac{|supp(R) \cap (E \setminus (\cup_{R' \in \mathcal{R}} supp(R')))|}{|E \setminus (\cup_{R' \in \mathcal{R}} supp(R'))|}$ measures the added instance coverage of a redescription R. $ARCS_{new}(R) = 1 - \frac{|attrs(R) \cap (V \setminus (\cup_{R' \in \mathcal{R}} attrs(R')))|}{|V \setminus (\cup_{R' \in \mathcal{R}} attrs(R'))|}$ measures the added attribute coverage of a redescription R. $ERCS_{new}(R)$ and $ARCS_{new}(R)$, by convention, have values 1 for all tested redescriptions when redescription set under development already covers all instances or attributes. Update as in equation 3 is denoted $mesU_1$ in Algorithm 1, as in equation 2 mes_1 and the optimisation function for redescription set optimisation listed at the bottom of Table 1 is denoted mes.

The procedure in Algorithm 1 constructs a redescription set of size m using steps from equations 1 and 3 containing coverage criteria (line 1, function `constructRS`). Next, it uses each redescription from this set as an initial (seed) redescription to construct a new candidate redescription set of size m, using a procedure defined in equation 3 (lines 7 - 9, function `compRS`). This allows reducing the error caused by the choice of a seed redescription. The original set plus the m newly created sets of size m are candidate sets in the local search procedure. All these sets are improved upon in the consecutive iterations. For each candidate set, we create a number of sets obtained by removing a predefined (decreasing) number of randomly selected redescriptions from the candidate set and completing these incomplete sets using procedure from equation 3 (lines $11-16$). A function `removeReds(`$\mathcal{R}$`, k)` removes k random redescriptions from a redescription set $\mathcal{R}$, whereas the function `compRS(`$\mathcal{R}$`, ...)` completes the incomplete set $\mathcal{R}$ to required predefined cardinality using procedures from equation 3. Next, we compare the candidate set to the newly created completed sets and if better set is found, replace the candidate with the best completed set (lines $17-18$). To evaluate a redescription set, we use the redescription set optimization function from Table 1. Candidate selection is repeated for a number of iterations, and the number of randomly removed redescriptions from candidate sets is reduced in each consecutive iteration. These steps reduce the error caused by the greedy nature of a redescription set construction procedure from [19].

5.1 Experiments

Algorithm 1 is applied to redescription sets produced by the CLUS-RM algorithm [19] on 5 datasets, the Country [8], the Phenotype [2, 20], the Mammals [6], the DBLP [6] and the Bio [18] datasets. Characteristics of these datasets are given in Table 2.

Algorithm 1 Local search procedure for redescription set optimization

Require: available redescriptions $\mathcal{R}_{all}$, output redescription set size $m \leq |\mathcal{R}_{all}|$, number of iterations $numit$, iteration change coefficient $itCh$, exploration strategy st
Ensure: $\mathcal{R}_{out}$, $|\mathcal{R}_{out}| = m$
$currIt \leftarrow numIt$, $\mathcal{R}_{mes} = constructRS(\mathcal{R}_{all}, m, mesU_1)$
$c \leftarrow \mathcal{R}_{mes}$, $indexSet \leftarrow \emptyset$
if $st == L$ **then** $indexSet_i \leftarrow (m-1) - i \cdot itCh,\ i = 0, \ldots, \lfloor \frac{m-1}{itCh} \rfloor$
else $indexSet_i \leftarrow (m-1)/itCh^i,\ i = 0, \ldots, \lfloor log_{itCh}(m-1) \rfloor$
end if
for $i \in indexSet$ **do**
  if $i == m-1$ **then** ▷ multi-threaded computation
    for $R \in \mathcal{R}_{mes}$ **do** $\mathcal{R}_{c_k} \leftarrow compRS(\{R\}, m, \mathcal{R}_{all}, mesU_1)$, $c \leftarrow c \cup \mathcal{R}_{c_k}$
    end for ▷ multi-threaded computation
  else
    for $\mathcal{R}_c \in c$ **do**
      $tmp \leftarrow \emptyset$
      for $j = 1, \ldots, currIt$ **do** ▷ multi-threaded computation
        $\mathcal{R}_{inc} \leftarrow removeReds(R_c, i)$
        $\mathcal{R}_{cmes} \leftarrow compRS(\mathcal{R}_{inc}, m, \mathcal{R}_{all}, mesU_1)$, $tmp \leftarrow tmp \cup \{\mathcal{R}_{cmes}\}$
      end for
      $\mathcal{R}_{mmes} \leftarrow arg\ min\ mes(\mathcal{R} \in tmp)$
      if $mes(\mathcal{R}_{mmes}) < mes(R_c)$ **then** $c \leftarrow c \setminus \{\mathcal{R}_c\} \cup \mathcal{R}_{mmes}$
      end if
    end for
  end if
  if $st == L$ **then** $currIt+ = itCh$
  else $currIt* = itCh$
  end if
end for
return $arg\ min_{\mathcal{R} \in c} mes(\mathcal{R})$

Table 2. TW_i denotes the attribute type of the i-th view (N - numeric, B - Boolean). ES_x denotes the approximate size of the exploration space when $|\mathcal{R}_{out}| = x$.

$\mathcal{D}$	$\lvert E\rvert$	$\lvert V\rvert$	T_{W_1}	T_{W_2}	$\lvert\mathcal{R}_{all}\rvert$	ES_{50}	ES_{100}	ES_{200}
Country	199	361	N	N	$[2936, 3262]$	$\approx 1.0 \cdot 10^{111}$	$\approx 3.4 \cdot 10^{191}$	$\approx 4.3 \cdot 10^{315}$
Mammals	2575	242	B	N	$[1969, 3054]$	$\approx 2.6 \cdot 10^{104}$	$\approx 2.2 \cdot 10^{170}$	$\approx 1.5 \cdot 10^{319}$
Phenotype	92	1230	N	N	$[407, 585]$	$\approx 4.9 \cdot 10^{70}$	$\approx 6.9 \cdot 10^{114}$	$\approx 1.2 \cdot 10^{121}$
DBLP	6455	6759	B	B	$[188, 292]$	$\approx 7.1 \cdot 10^{56}$	$\approx 1.5 \cdot 10^{55}$	$\approx 1.0 \cdot 10^{61}$
Bio	3475	4523	N	N	$[2073, 3412]$	$\approx 1.0 \cdot 10^{112}$	$\approx 4.7 \cdot 10^{172}$	$\approx 1.3 \cdot 10^{284}$

5.2 Algorithmic parameters

In all experiments, we used Predictive Clustering regression trees of depth 8. In addition to the main search guiding trees, we used a supplementary random for-

est with 6 trees. We utilized redescription query minimization, the redescription refinement procedure and allowed the use of logical conjunction, disjunction and negation operator to construct redescription queries. Maximum redescription support set size was set to $\approx 0.8 \cdot |E|$ on all datasets. Minimal redescription support size was set to 5 on the Country, Phenotype (dataset with small $|E| < 200$) and the DBLP dataset (sparse data) and on 10 on the Mammals dataset (dataset with medium-sized $|E| < 5000$) and the Bio dataset ($|E| < 4000$). Maximal redescription p-value of 0.01 was used in all experiments. The minimal redescription Jaccard index of 0.7 was used on the Bio dataset, 0.5 was used on the Country, the Phenotype and the Mammals dataset and 0.3 on the sparse DBLP dataset. We used 5 random restarts (initializations) on the Country and the DBLP datasets, 6 on the Phenotype dataset and 2 on the Mammals and the Bio dataset. 20 iterations per a restart was used on the Country, 50 on the Phenotype, 5 on the DBLP and the Bio and 2 on the Mamamls dataset. Scores of $\mathcal{R}_{out}$ produced from $\mathcal{R}_{all}$, as described in [19], where best redescription selection in each iteration is performed following the procedure from [19] (denoted s_{org}) and a modified procedure from equation 3 (denoted s_0), are compared to scores of $\mathcal{R}_{out}$ constructed using Algorithm 1 for output redescription sets of size 50 and 100. Experiments for different output set sizes are independent. Iteration change coefficient of 4 was used with $st \neq L$ and $numIt = 10$.

To approximate the ideal objective vector, we first computed the m most accurate, the m most significant and m redescriptions with the smallest complexity from $\mathcal{R}_{all}$. This gives us the ideal value for AJS, ApS and $AcompS$. $AEJS$, $AAJS$, $ERCS$ and $ARCS$ are approximated using redescription set construction with preference weights $(0.01, 0.01, 0.94, 0.01, 0.01, 0.01, 0.01)$, $(0.01, 0.01, 0.01, 0.94, 0.01, 0.01, 0.01)$, $(0.01, 0.01, 0.01, 0.01, 0.01, 0.94, 0.01)$ and $(0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.94)$. Non-zero weights are used to preserve the property of obtaining Pareto optimal solution at each step of the procedure.

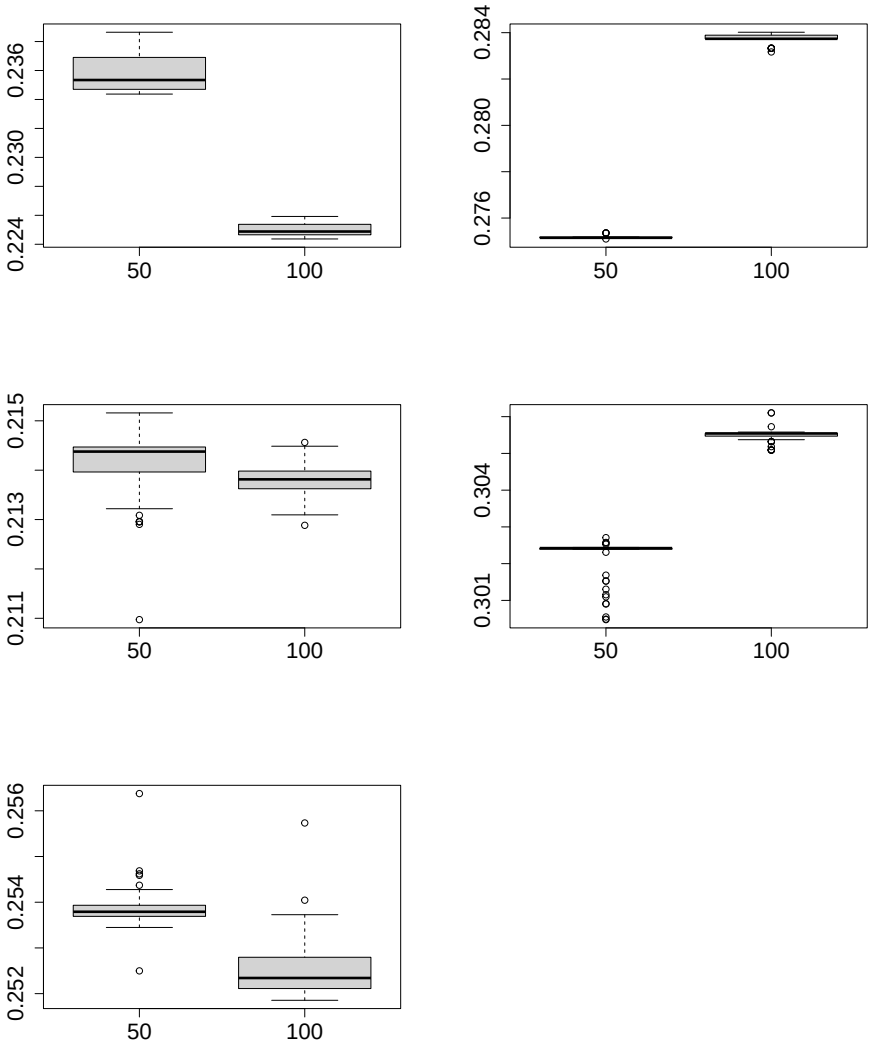
5.3 Results

Figure 1 demonstrates the variability in final redescription set scores when different seed redescriptions are used to create output redescription sets. Measured variability is relatively small (< 0.004) demonstrating present, but relatively moderate influence of a seed redescription on a final redescription set quality.

Finally, we report the score of the best redescription set obtained in each iteration of Algorithm 1, starting from the redescription set obtained by the greedy approach from [19]. The value of the approximation of the ideal objective vector given $\mathcal{R}_{all}$ is denoted m_{iov} and the difference of the best obtained solution to the ideal objective vector is denoted $d_{best,iov}$. The initial starting set s_0 used in Algorithm 1 equals the set obtained by the greedy procedure from [19], where the best redescription is computed using a modified procedure from equation 3.

- Country
  - $|\mathcal{R}_{out}| = 50:\ 0.2383_{s_{org}},\ 0.2346_{s_0},\ 0.2344_{s_1},\ 0.2335_{s_2}, 0.2329_{s_3},\ m_{\text{iov}} = 0.1709,\ d_{best,iov} = 0.062.$

Fig. 1. Distributions of the *mes* score of the m candidate sets obtained by utilizing each redescription from $\mathcal{R}_{mes}$ as a seed redescription on the Country (top left), the Phenotype (top right), the Mammals (middle left), the DBLP (middle right) and the Bio (bottom left) datasets. Measurements are performed for output redescription sets of cardinality 50 and 100.



- $|\mathcal{R}_{out}| = 100:\ 0.2330_{s_{org}},\ 0.2247_{s_0}, 0.2244_{s_1}, 0.2236_{s_2}, 0.2232_{s_3}, 0.2231_{s_4}$, $m_{\text{iov}} = 0.1476$, $d_{best,iov} = 0.076$.

– Mammals

- $|\mathcal{R}_{out}| = 50:\ 0.2189,\ 0.2144_{s_0},\ 0.2110_{s_1},\ 0.2104_{s_2},\ 0.2097_{s_3},\ m_{\text{iov}} = 0.1142,\ d_{best,iov} = 0.096.$
- $|\mathcal{R}_{out}| = 100:\ 0.2214_{s_{org}},\ 0.2140_{s_0}, 0.2129_{s_1}, 0.2117_{s_2}, 0.2112_{s_3}, 0.2111_{s_4},\ m_{\text{iov}} = 0.1325,\ d_{best,iov} = 0.079.$

– Phenotype
- $|\mathcal{R}_{out}| = 50:\ 0.2908_{s_{org}},\ 0.2751_{s_0},\ 0.2751_{s_1},\ 0.2749_{s_2}, 0.2748_{s_3},\ m_{\text{iov}} = 0.2283,\ d_{best,iov} = 0.047.$
- $|\mathcal{R}_{out}| = 100:\ 0.2917_{s_{org}},\ 0.2839_{s_0}, 0.2832_{s_1}, 0.2831_{s_2}, 0.2831_{s_3}, 0.2831_{s_4},\ m_{\text{iov}} = 0.2380,\ d_{best,iov} = 0.045.$

– DBLP
- $|\mathcal{R}_{out}| = 50:\ 0.3196_{s_{org}},\ 0.3024_{s_0},\ 0.3005_{s_1},\ 0.3002_{s_2}, 0.2999_{s_3},\ m_{\text{iov}} = 0.2204,\ d_{best,iov} = 0.080.$
- $|\mathcal{R}_{out}| = 100:\ 0.3096_{s_{org}},\ 0.3055_{s_0}, 0.3051_{s_1}, 0.3043_{s_2}, 0.3039_{s_3}, 0.3039_{s_4},\ m_{\text{iov}} = 0.2565,\ d_{best,iov} = 0.047.$

– Bio
- $|\mathcal{R}_{out}| = 50:\ 0.2465_{s_{org}},\ 0.2538_{s_0},\ 0.2525_{s_1},\ 0.2519_{s_2}, 0.2516_{s_3},\ m_{\text{iov}} = 0.1704,\ d_{best,iov} = 0.073.$
- $|\mathcal{R}_{out}| = 100:\ 0.2487_{s_{org}},\ 0.2535_{s_0}, 0.2519_{s_1}, 0.2515_{s_2}, 0.2510_{s_3}, 0.2509_{s_4},\ m_{\text{iov}} = 0.1757,\ d_{best,iov} = 0.072.$

The presented experiments show that the local search procedure outperforms the greedy procedure from [19] on 4 out of 5 datasets. The biggest improvement was achieved on the DBLP dataset with output size of cardinality 50 equalling 1.97% od optimization function value range (relative improvement of 6.6%). Improvement (decrease of redescription set score) is noticeable in each iteration of Algorithm 1 on every dataset, indicating the ability of local search procedure to correct the aforementioned errors of the greedy procedure. The largest increase in output redescription set quality during the local search, performed using Algorithm 1 was $\approx 0.47\%$ of the function value range and the relative improvement of $\approx 2.24\%$. It was achieved on the Mammals dataset during the construction of the output redescription set of cardinality 50. The difference to the ideal objective vector demonstrates there could still be place for improvement on all studied datasets with the maximum potential improvement of up to 9.6% of the objective function value range on the Mammals dataset.

Source code of the local search procedure can be found on Github: https://github.com/matmih/RMLocalSearch.git.

6 Conclusions and future work

This work shows that the proposed local search procedure for fixed-size redescription set construction can improve upon the existing greedy procedure on 4 out of 5 used datasets. The main two sources of error of the existing greedy approach for fixed size redescription set construction: a) the choice of the seed redescription and b) the greedy nature of the candidate selection, can be reduced using adapted local search procedure. We have shown that output redescription sets can be improved on each of the 5 utilised datasets, although obtained improvements are

moderate. However, there is still room for improvement as demonstrated by the ideal objective vector computation, which forms a lower bound of the obtainable error given the predefined cardinality of the output redescription sets and the available pool of redescriptions $\mathcal{R}_{all}$.

The main drawback of the proposed procedure is the requirement to generate and evaluate relatively high number of candidate redescription sets (depending on algorithmic parameters), which can be time consuming. Fortunately, the overall execution times can be significantly reduced using parallel execution.

Future work should focus on adapting more advanced space exploration techniques for fixed size redescription set optimisation. Some of the prominent candidates are simulated annealing, evolutionary optimisation, bio-inspired optimisation algorithms, neural networks for combinatorial optimisation and increasingly successful generative deep networks and ultimately the foundation models.

References

1. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A.I.: Fast discovery of association rules. In: Advances in Knowledge Discovery and Data Mining, pp. 307–328. American Association for Artificial Intelligence (1996)
2. Brbić, M., Piškorec, M., Vidulin, V., Kriško, A., Šmuc, T., Supek, F.: The landscape of microbial phenotypic traits and associated genes. Nucleic Acids Research **44**(21), 10074–10090 (10 2016). https://doi.org/10.1093/nar/gkw964
3. Chisholm, M., Tadepalli, P.: Learning decision rules by randomized iterative local search. In: Machine Learning, Proceedings of the Nineteenth International Conference (ICML 2002), University of New South Wales, Sydney, Australia, July 8-12, 2002. pp. 75–82. Morgan Kaufmann (2002)
4. Feige, U., Mirrokni, V.S., Vondrák, J.: Maximizing non-monotone submodular functions. SIAM Journal on Computing **40**(4), 1133–1153 (2011). https://doi.org/10.1137/090779346
5. Fournier-Viger, P., Lin, J.C., Vo, B., Truong, T.C., Zhang, J., Le, H.B.: A survey of itemset mining. WIREs Data Mining Knowledge Discovery **7**(4) (2017). https://doi.org/10.1002/WIDM.1207
6. Galbrun, E.: Methods for Redescription Mining. Ph.D. thesis, University of Helsinki, Finland (2013)
7. Galbrun, E., Miettinen, P.: From black and white to full color: extending redescription mining outside the boolean world. Statistical Analysis and Data Mining **5**(4), 284–303 (2012). https://doi.org/10.1002/SAM.11145
8. Gamberger, D., Mihelčić, M., Lavrač, N.: Multilayer clustering: A discovery experiment on country level trading data. In: Proceedings of the 17th International Conference on Discovery Science (DS 2014), Bled. pp. 87–98. Springer International Publishing (2014). https://doi.org/{10.1007/978-3-319-11812-3_8}
9. Gunantara, N.: A review of multi-objective optimization: Methods and its applications. Cogent Engineering **5**(1), 1502242 (2018). https://doi.org/10.1080/23311916.2018.1502242
10. Jabba, A.M.: Rule induction with iterated local search. International Journal of Intelligent Engineering & Systems **14**(4) (2021)
11. Jaccard, P.: The distribution of the flora in the alpine zone. 1. New phytologist **11**(2), 37–50 (1912)

12. Jacques, J., Taillard, J., Delerue, D., Jourdan, L., Dhaenens, C.: Multi-objective local search for mining pittsburgh classification rules. In: International Conference on Metaheuristics and Nature Inspired Computing (2012)
13. Kalofolias, J., Galbrun, E., Miettinen, P.: From sets of good redescriptions to good sets of redescriptions. Knowledge and Information Systems **57**(1), 21–54 (2018). https://doi.org/10.1007/S10115-017-1149-7
14. Kanamori, K.: Learning locally interpretable rule ensemble. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 360–377. Springer (2023)
15. Klösgen, W.: Explora: A multipattern and multistrategy discovery assistant. In: Advances in Knowledge Discovery and Data Mining, pp. 249–271. American Association for Artificial Intelligence, Menlo Park, USA. (1996)
16. Lakkaraju, H., Bach, S.H., Leskovec, J.: Interpretable decision sets: A joint framework for description and prediction. In: Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining (KDD 2016), San Francisco. pp. 1675–1684. ACM (2016). https://doi.org/10.1145/2939672.2939874
17. Lavrač, N., Kavšek, B., Flach, P., Todorovski, L.: Subgroup discovery with cn2-sd. Journal of Machine Learning Research **5**, 153–188 (2004)
18. Mihelčić, M., Šmuc, T., Supek, F.: Patterns of diverse gene functions in genomic neighborhoods predict gene function and phenotype. Scientific Reports **9**(1), 19537 (2019)
19. Mihelčić, M., Džeroski, S., Lavrač, N., Šmuc, T.: A framework for redescription set construction. Expert Systems with Applications **68**, 196–215 (2017). https://doi.org/10.1016/J.ESWA.2016.10.012
20. Mihelčić, M., Šmuc, T.: Targeted and contextual redescription set exploration. Machine Learning **107**(11), 1809–1846 (2018). https://doi.org/10.1007/S10994-018-5738-9
21. Ramakrishnan, N., Kumar, D., Mishra, B., Potts, M., Helm, R.F.: Turning CARTwheels: An alternating algorithm for mining redescriptions. In: Proceedings of the 10th International Conference on Knowledge Discovery and Data Mining (KDD 2004), Seattle. pp. 266–275. ACM (2004). https://doi.org/10.1145/1014052.1014083
22. Reynolds, A.P., de la Iglesia, B.: A multi-objective GRASP for partial classification. Soft Comput. **13**(3), 227–243 (2009). https://doi.org/10.1007/S00500-008-0320-1, https://doi.org/10.1007/s00500-008-0320-1
23. Szczepanski, N., Audemard, G., Jourdan, L., Lecoutre, C., Mousin, L., Veerapen, N.: A hybrid CP/MOLS approach for multi-objective imbalanced classification. In: GECCO '21: Genetic and Evolutionary Computation Conference, Lille, France, July 10-14, 2021. pp. 723–731. ACM (2021). https://doi.org/10.1145/3449639.3459310, https://doi.org/10.1145/3449639.3459310
24. Wrobel, S.: An algorithm for multi-relational discovery of subgroups. In: Principles of Data Mining and Knowledge Discovery, Lecture Notes in Computer Science, vol. 1263, pp. 78–87. Springer, Berlin / Heidelberg (1997). https://doi.org/10.1007/3-540-63223-9_108