Action Rule induction by Sequential Covering

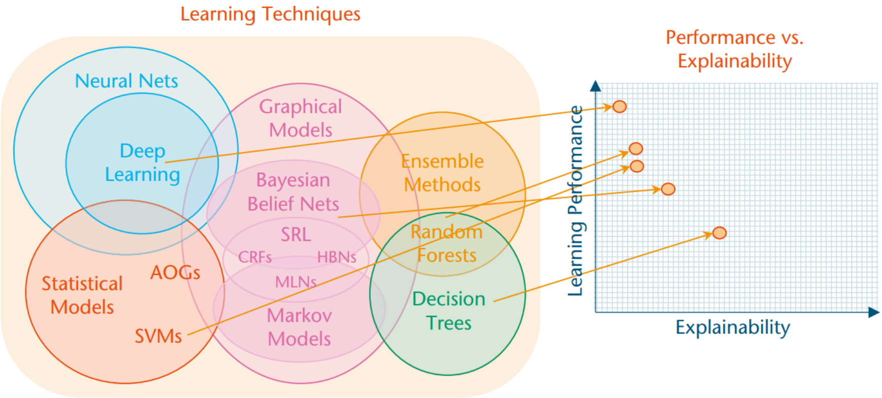
Paweł Matyszok

Department of Computer Networks and System
Silesian University of Technology

Agenda

Agenda

Explainability vs. Performance trade-off



Gunning, D. (2017). Explainable artificial intelligence (xai). Defense Advanced Research Projects Agency (DARPA)

Actionability

A pattern is actionable, if the user can take an action based on the pattern and benefit from it.

Action Rules are readable form of representation of Actionable Knowledge.

Agenda

Information System

Let us call following tuple an ,,Information System":
$\mathbb{A} = (U, A)$
where:

- U - Universe - set of objects
- A - Set of attributes, that describe objects in U

We can understand Information System (IS) as a table, where rows are depicting objects and columns are depicting values of attributes.
Distinguished attribute $d, d \in A$ is called decision attribute - the class of the object.

Decision rule

A logical formulae in the form of:

$$a_1 = v_{a_1} \wedge a_2 = v_{a_2} \wedge \cdots \wedge a_n = v_{a_n} \rightarrow d = v_d$$

where

- $a_k \in A$ - attributes
- $v_{a_k} \in V_{a_k}$ - values of particular attribute
- v_d - value of decision attribute

are called Decision Rules.
Simplified notation:

$$w_1 \wedge w_2 \wedge \ldots \wedge w_k \textbf{ THEN } d = v$$

Part on the left of $\rightarrow$ sign (or word **THEN**) is called premise, while condition on the right side is called conclusion or decision of the rule.

Elementary conditions

Subformulae $a_k = v_{a_k}$ are called elementary conditions. For numerical attributes elementary conditions can take many forms:

- $a_k \in (v_1, v_2)$
- $a_k \leq v_1$
- $a_k > v_1$
- ...

For nominal, discreet attributes, there is only one generic form, $a_k = v_1$.

Agenda

Two decision rules:

r1: $w_{1_1} \land w_{1_2} \land \ldots \land w_{1_k}$ **THEN** $d = v_1$

r2: $w_{2_1} \land w_{2_2} \land \ldots \land w_{2_k}$ **THEN** $d = v_2$

could be assembled into formula

$$w_{1_1} \rightarrow w_{2_1} \land w_{1_2} \rightarrow w_{2_2} \land \ldots \land w_{1_k} \rightarrow w_{2_k} \textbf{ THEN } d = v_1 \rightarrow v_2 \tag{1}$$

that we will call **Action Rule**.
Simplified notation:

r: $(a_1, v_{a_{11}} \rightarrow v_{a_{12}}) \land (a_2, v_{a_{21}} \rightarrow v_{a_{22}}) \land \ldots \land (a_k, v_{a_{k1}} \rightarrow v_{a_{k2}})$ **THEN** $(d = v_1 \rightarrow v_2)$

Action and meta-action

The premise of the action rule can contain:

- simple elementary conditions (a_k, v_k),
- elementary actions $(a_k, v_{k_1} \rightarrow v_{k_2})$
- narrowing actions $(a_k, ANY \rightarrow v_{k_2})$

Actions itself inform us about necessity to change the value of the attribute. The information about how to execute such change are called meta-actions.

Stable and flexible attributes

When inducing action rule, we might need to further divide attributes based on the technical possibility of implementing a change. We will consider:

- Stable attributes - no actions can be defined, only elementary conditions, i.e. date of birth, height
- Flexible attributes - able to be subject of an action, i.e. interest rate, particle concentration, room temperature

Agenda

Credit risk score

Based on the data from ,,German credit" dataset, we can suggest how someone can change their risk, as seen by a banking industry:

r1: $(credit\_amount, (3907.0, \infty) \rightarrow (1221.0, 3912.0)) \wedge (age, (22.0, \infty) \rightarrow (25.5, \infty))$
THEN $(class, bad \rightarrow good)$

Credit risk score II

r2: $(duration, (9.0, \infty) \to (3.0, \infty)) \land$
$(credit\_amount, (608.5, \infty) \to (213.0, 7826.5)) \land$
$(checking\_status, < 0 \to nochecking) \land$
$(existing\_credits, \langle 1.0, \infty) \to \langle 0.5, \infty)) \land$
$(age, \langle 16.0, \infty) \to (23.0, \infty))$
THEN $(class, bad \to good)$

The dataset features hidden business rule
IF $attr1 = attr2 \vee attr5 = 1$ **THEN** $class = 1$.

Selection of discovered rules:

r3: $(attr5, 4 \rightarrow 1) \wedge (attr1, 1)$ **THEN** $(class, 0 \rightarrow 1)$

r4: $(attr5, 4 \rightarrow 1)$ **THEN** $(class, 0 \rightarrow 1)$

r5: $(attr5, 3 \rightarrow 1)$ **THEN** $(class, 0 \rightarrow 1)$

Regression

Body fat percentage estimation

r4: IF $(\mathit{Forearm}, (-\infty, 29.15) \to \langle 26.85, \infty)) \wedge$
$(\mathit{Thigh}, (-\infty, 66.25) \to \langle 53.55, \infty)) \wedge$
$(\mathit{Biceps}, \langle 28.25, \infty) \to) \wedge$
$(\mathit{Density}, (-\infty, 1.05) \to \langle 1.06, \infty)) \wedge$
$(\mathit{Age}, \langle 27.50, 53) \to \langle 42.50, \infty)) \wedge$
$(\mathit{Weight}, \to \langle 155.13, \infty))$ **THEN**
$(\mathit{class}, 22.50 \pm 5.65 \to 13.80 \pm 2.93)$

Agenda

Problem statement

Currently existing methods of Action Rule induction have some flaws, including:

- Requirement of prior induction of decision rules,
- Lack of ability to work with continuous or missing data,
- Induction of very large sets of rules,
- No publicly available implementations

Research

Usage of Sequential Covering (SC) paradigm and classification rule quality measures to supervise induction of Action Rules could lead to concise and comprehensible rulesets.

Sequential Covering approach:

- has been proven effective in decision rules induction,
- is simple to understand and implement,
- can serve as basis for beam-search

Contribution

My work includes:

- Introducing first algorithm and program to discover Action Rules using SC paradigm called F-ARI (Forward Action Rule Induction)
- Introduction of Backward-ARI (B-ARI) method, that allows to discover interesting ARs for some class of problems and ensemble of F-ARI and B-ARI methods
- Modification of ARI method to support also regression data
- Creation of method to resolve conflicts between action rules and induction of recommendations
- Creation of framework to assess quality of action rulesets and recommendations

Sequential Covering Action Rule Induction

Input: $E(A, \{d\})$—training data set, *mincov*—minimum number of yet uncovered examples that a new rule must cover, C_S, C_T—Source and Target classes, Q—rule quality measure

Output: R—action rule set.

$E_U \leftarrow E_S$ ▷ set of uncovered source-class examples

$R \leftarrow \emptyset$ ▷ start from an empty rule set

repeat

  $r \leftarrow \emptyset \rightarrow C_S \rightarrow C_T$ ▷ start from an empty premise with known conclusion

  $r \leftarrow \textsc{GrowActionRule}(r, E, E_U, \mathit{mincov}, Q)$ ▷ grow actions

  $r \leftarrow \textsc{PruneActionRule}(r, E, Q)$ ▷ prune actions

  $R \leftarrow R \cup \{r\}$

  $E_U \leftarrow E_U \setminus \textsc{Cov}(r, E_U)$ ▷ remove from E_U examples covered by source of r

until $|E_U| < \mathit{mincov}$

Rule induction - example of rule growth

iteration	w_{best_S}	q_{r_S}	w_T	q_{r_T}
1	$(a_1 = 1)$	0.69	$(a_1 = 3)$	0.70
2	$(a_2 = 2)$	0.88	$(a_2 = 3)$	1.00
3	$(a_6 = 2)$	0.90	$(a_6 = 2)$	1.00

Consecutive source and target parts of elementary actions induced during the action rule growing (q - rule precision) on Monk1 dataset.

IF $((a_1 = 1) \rightarrow (a_1 = 3)) \wedge$
$((a_2 = 2) \rightarrow (a_2 = 3)) \wedge$
$((a_6 = 2) \rightarrow (a_6 = 2))$
THEN $(class = 0) \rightarrow (class = 1)$

Rule induction - pruning example

rule premise	q_{r_S}	q_{r_T}
$(a_1 = 1) \rightarrow (a_1 = 3) \wedge (a_2 = 2) \rightarrow (a_2 = 3)) \wedge (a_6 = 2) \rightarrow (a_6 = 2)$	0.13	0.27
$(a_1 = 1) \rightarrow (a_1 = 3) \wedge (a_2 = 2) \rightarrow (a_2 = 3) \wedge (a_6 = 2) \rightarrow$	0.13	0.27
$(a_1 = 1) \rightarrow (a_1 = 3) \wedge (a_2 = 2) \rightarrow (a_2 = 3)$	0.21	0.27
$(a_1 = 1) \rightarrow (a_1 = 3) \wedge (a_2 = 2) \rightarrow$	0.21	0.24
$(a_1 = 1) \rightarrow (a_1 = 3) \wedge \rightarrow (a_2 = 3)$	0.26	0.27

Steps taken to prune the action rule (q - RSS)

IF $((a_1 = 1) \rightarrow (a_1 = 3)) \wedge (\rightarrow (a_2 = 3))$
THEN $(class, 0) \rightarrow (class, 1)$

Bidirectional action rule learning

- Forward induction starts search in source class:
  $(a_1, v_{1_s} \rightarrow v_{1_t}) \wedge \ldots \rightarrow (d, s \rightarrow t)$
- Backward method starts the search among examples of target class, building the rule contrary to demands of the user:
  $(a_1, v_{1_t} \rightarrow v_{1_s}) \wedge \ldots \rightarrow (d, t \rightarrow s)$ and then attempts to revert the rule.
- Union of forward and backward rules were proven to be more effective in conducted experiments.

Recommendations

- SC method leads to contradicting rulesets, where singular example might be covered by more then one rule.
- If actions from the ruleset are to be applied on objects, the conflict must be resolved.
- Work on conflict resolution lead to creation of recommendations, which are specialized action rules created with single example in mind. A new algorithm was proposed.

Recommendations from action rulesets

Input: R—Action Ruleset,
Output: MT—Metatable

$T[] \leftarrow$ $\textsc{GetConditionsGroupedByAttribute}(R)$ $\triangleright$ T[i] contains all elementary conditions for i-th attribute from rules in R
for $i = 0, i < n$ **do** $\triangleright$ n is the number of attributes
  $T[i] \leftarrow \textsc{EliminateIntersections}(T[i])$
  $i \leftarrow i + 1$
$MT \leftarrow \textsc{CartesianProduct}(T)$
return MT

Metatable - data structure for recommendations

id	${}_m a_1$	${}_m a_2$	$\dots$	${}_m a_n$
1	Va_{1_1}	Va_{2_1}	$\dots$	Va_{n_1}
2	Va_{1_2}	Va_{2_2}	$\dots$	Va_{n_2}
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$

Recommendation discovery

Input: MT—Metatable, $E(A, d)$—Training dataset, e—analyzed example, Q—rule quality measure, C_S, C_T— source and target class
Output: r—Recommendation

$S \leftarrow (x \in MT[i] : \textsc{Cov}(x, e)) \rightarrow C_S$ ▷ row from MT that covers example e
$T \leftarrow \emptyset \rightarrow C_T$
repeat
    $w_{best} \leftarrow \textsc{GetBestElementaryCondition}(MT, T, Q, C_S, C_T)$
    $T \leftarrow T \wedge w_{best}$
    ${}_m a \leftarrow \textsc{GetAttribute}(w_{best})$
    $MT \leftarrow MT \setminus {}_m a$
until $w_{best} = \emptyset$
$r \leftarrow (S \rightarrow T)$
return r

Recommendation discovery - example

Considering following ruleset:

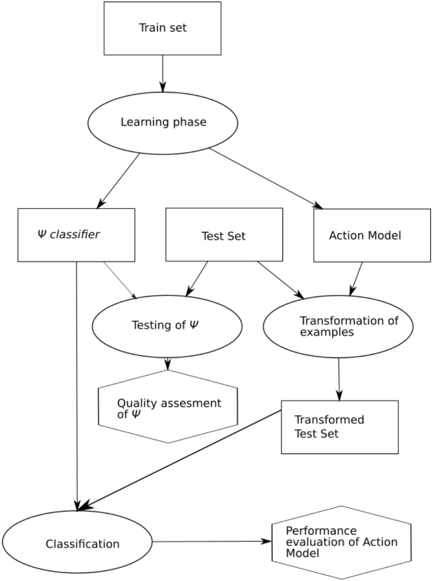
r1: IF $((body\ temperature > 38^{\circ}C) \rightarrow (body\ temperature < 36.6^{\circ}C)) \wedge$ $((pus\ on\ tonsils = Yes) \rightarrow (pus\ on\ tonsils = No))$
THEN $(ill\ = Yes) \rightarrow (ill\ = No)$

r2: IF $((body\ temperature > 37.5^{\circ}C) \rightarrow (body\ temperature < 37^{\circ}C)) \wedge$ $((pus\ on\ tonsils = No))$
THEN $(ill\ = Yes) \rightarrow (ill\ = No)$

Recommendation discovery - example metatable

body temperature	*pus on tonsils*
$(min, 36.6]$	*No*
$(min, 36.6]$	*Yes*
$(36.6, 37]$	*No*
$(36.6, 37]$	*Yes*
$(37, 37.5]$	*No*
$(37, 37.5]$	*Yes*
$(37.5, 38]$	*No*
$(37.5, 38]$	*Yes*
$(38, max)$	*No*
$(38, max)$	*Yes*

Quality assessment framework



Action Model

Strategies of selection of an action rule to be applied:

1. Use recommendation method

2. Select best action rule from action ruleset using one of the quality measures

Proposed quality metrics of Action Models

For classification data:

m_1: $\frac{\text{count of examples for which action was not provided}}{\text{count of source class examples in test set}}$

m_2: $\frac{\text{count of examples classified as target class}}{\text{count of examples, for which action was provided}}$

m_3: $\frac{\text{count of examples classified as source class}}{\text{count of examples, for which action was provided}}$

For regression data:

mr_1: $\mathrm{RMSE}(v'_d, \bar{v_d})$

mr_2: $\mathrm{MAE}(v'_d, \bar{v_d})$

mr_3: $\frac{|\{\bar{v_{d_i}} \in (v'_{d_i} - s(v'_d), v'_{d_i} + s(v'_d))\}|}{\text{count of examples, for which action was provided}}$

F-ARI rules characteristics

	C2	Correlation	Information Gain	RSS	Weighted Laplace
#rules	11.86	5.62	6.00	3.87	14.19
#elementary conditions	3.62	3.46	3.45	3.23	3.32
#elementary actions	1.81	1.35	1.43	1.17	1.85
source precision	0.93	0.83	0.84	0.78	0.94
target precision	0.93	0.85	0.86	0.83	0.94
source coverage	0.43	0.62	0.62	0.68	0.31
target coverage	0.41	0.63	0.60	0.66	0.29

Selected characteristics of rulesets discovered with F-ARI method. Values averaged over 16 test datasets.

B-ARI rules characteristics

	C2	Correlation	Information Gain	RSS	Weighted Laplace
#rules	15.43	5.21	6.38	4.38	19.82
#elementary conditions	3.52	3.35	3.37	3.23	3.25
#elementary actions	2.89	2.57	2.65	2.37	2.71
source precision	0.92	0.84	0.84	0.79	0.91
target precision	0.93	0.85	0.86	0.81	0.95
source coverage	0.44	0.64	0.64	0.71	0.34
target coverage	0.43	0.61	0.60	0.66	0.32

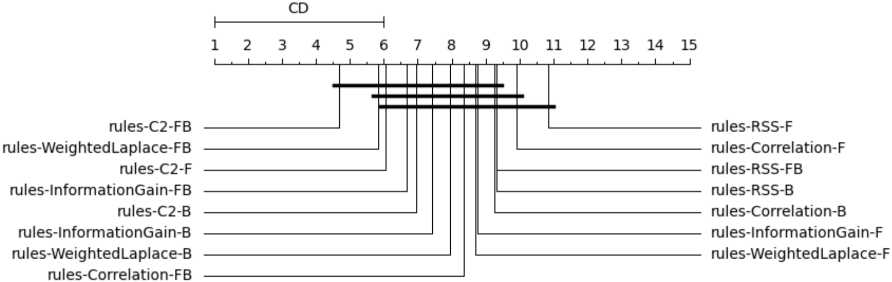
Selected characteristics of rulesets discovered with B-ARI method. Values averaged over 16 test datasets.

Recommendation accuracy

recommendation accuracy	Method	precision	wLap	C2	Gain	Corr	RSS
the best action rule	Forward	66.6	69.5	69.7	62.1	62.2	61.0
the best action rule	Backward	50.0	55.0	63.6	67.9	63.8	59.9
recommendation	Forward	60.1	74.3	82.1	79.5	78.2	75.7
recommendation	Backward	62.8	75.7	85.7	82.8	81.8	76.8

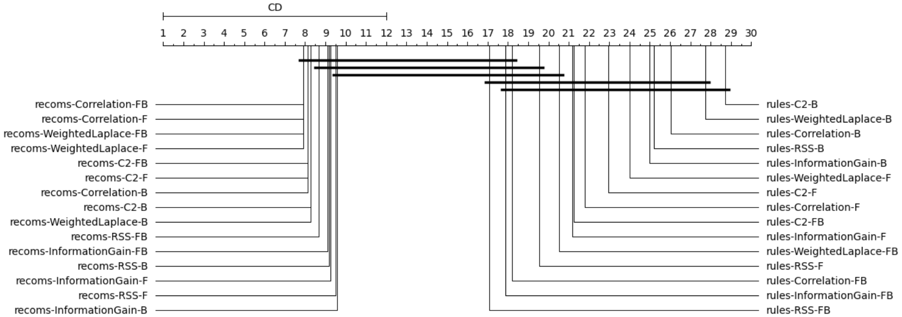
Recommendation accuracy m_2. The results are given as a percentage. Data averaged on 16 test datasets.

Ensemble of Forward and Backward rules



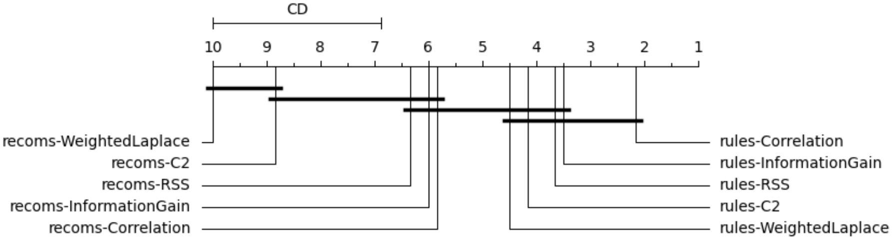
CD-diagram indicating differences in m_2 achieved by various rule-based action models

Coverage of unknown examples



CD-diagram indicating differences in ability to cover new examples by recommendation and rule-based algorithms.

Quality of methods applied to regression data



CD-diagram showing differences in performance measured by mr_2 metric of various methods on regression data (reversed scale).

Quality assessment results - highlights

Conducted experiments show that:

- On classification data highest score was achieved by models trained with C2 and WLap functions;
- On regression data best results are achieved for models trained with C2 function;
- Fusion of Forward and Backward rules leads to increased fidelity of the model;
- Recommendation method outperforms rulesets in coverage of previously unknown source class examples.

Publications

M. Kozielski, P. Matyszok, M. Sikora and Ł. Wróbel, *Decision rule learning from stream of measurements - a case study in methane hazard forecasting in coal mines* in Man-machine interactions 5, ICMMI 2017

P. Matyszok, M. Sikora and Ł. Wróbel, *Covering approach to action rule learning* in Beyond databases, architectures and structures : Facing the challenges of data proliferation and growing variety. 14th International conference, BDAS 2018 held at the 24th IFIP World Computer Congress

P. Matyszok, Ł. Wróbel and M. Sikora, *Bidirectional action rule learning* in Computer and information sciences : 32nd International symposium, ISCIS 2018 held at the 24th IFIP World Computer Congress

M. Sikora, P. Matyszok and Ł. Wróbel, *SCARI: Separate and conquer algorithm for action rules and recommendations induction* in Information Sciences, Volume 607, 2022, Pages 849-868
https://github.com/adaa-polsl/SCARI

Q/A

Thank you for your attention!
Any questions?

Agenda

Action Rule Specialization I

Input: r—input action rule, E—training data set, E_U—set of examples uncovered by source of r, mincov—minimum number of previously uncovered examples that a new rule must cover.

Output: r—grown rule.

function $\textsc{GrowActionRule}(r, E, E_U, mincov)$

  $r_S \leftarrow \textsc{GetSourcePart}(r)$

  $r_T \leftarrow \textsc{GetTargetPart}(r)$

  $q_{\mathrm{best}_S} \leftarrow -\infty, \quad \mathrm{cov}_{\mathrm{best}_S} \leftarrow -\infty$ ▷ best quality, coverage of source

  $q_T \leftarrow -\infty, \quad \mathrm{cov}_T \leftarrow -\infty$ ▷ best quality, coverage of target

  repeat

    $w_{\mathrm{best}_S} \leftarrow \emptyset$ ▷ current source best condition

    $w_T \leftarrow \emptyset$ ▷ current target condition

    $E_r \leftarrow \textsc{Cov}(r_S, E)$ ▷ examples from E satisfying r_S premise

    for $w \in \textsc{GetPossibleConditions}(E_r)$ **do**

      $r_{S_w} \leftarrow r_S \land w$ ▷ source rule extended with condition w

$E_{r_{S_w}} \leftarrow \textsc{Cov}(r_{S_w}, E)$

if $|E_{r_{S_w}} \cap E_U| \geq \mathit{mincov}$ **then** $\triangleright$ verify coverage requirement

$\quad q \leftarrow \textsc{Quality}(E_{r_{S_w}},\ E \setminus E_{r_{S_w}})$ $\triangleright$ rule quality measure

$\quad$ **if** $q > q_{\mathrm{best}_S}$ **or** $(q = q_{\mathrm{best}_S}$ **and** $|E_{r_{S\,w}}| > \mathrm{cov}_{\mathrm{best}_S})$ **then**

$\qquad w_{\mathrm{best}_S} \leftarrow w, \quad q_{\mathrm{best}_S} \leftarrow q, \quad \mathrm{cov}_{\mathrm{best}_S} \leftarrow |E_{r_{S_w}}|$

$E_r \leftarrow \textsc{Cov}(r_T,\ E)$ $\triangleright$ examples from E satisfying r_T premise

$a \leftarrow \textsc{GetAttribute}(w_{\mathrm{best}_S})$

for $w \in \textsc{GetPossibleConditionsForAttribute}(E_r, a)$ **do**

$\quad r_{T_w} \leftarrow r_T \wedge w$

$\quad E_{r_{T\,w}} \leftarrow \textsc{Cov}(r_{T_w}, E)$

$\quad$ **if** $|E_{r_{T\,w}}| \geq \mathit{mincov}$ **then** $\triangleright$ verify coverage requirement

$\qquad q \leftarrow \textsc{Quality}(E_{r_{T\,w}},\ E \setminus E_{r_{T\,w}})$ $\triangleright$ rule quality measure

$\qquad$ **if** $q > q_T$ **or** $(q = q_T$ **and** $|E_{r_{T\,w}}| > \mathrm{cov}_T)$ **then**

$\qquad\quad w_T \leftarrow w, \quad q_T \leftarrow q, \quad \mathrm{cov}_T \leftarrow |E_{r_{T\,w}}|$

Action Rule Specialization III

$r \leftarrow r \wedge (w_{\text{best}_S} \rightarrow w_T)$ ▷ Extend rule with new elementary action

until $w_{\text{best}_S} = \emptyset$

return r