Meta-Learning Rule Learning Heuristics

Frederik Janssen and Johannes Fürnkranz

Technische Universität Darmstadt
Fachbereich Informatik
Fachgebiet Knowledge Engineering
{janssen,juffi}@ke.informatik.tu-darmstadt.de

Outline

Separate-and-Conquer Rule Learning

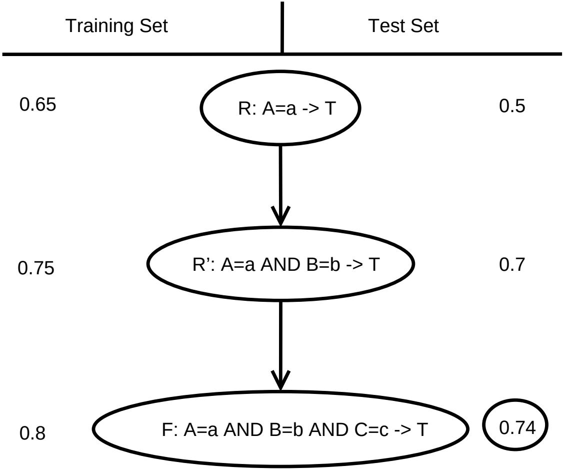
- we used a standard Separate-and-Conquer algorithm for our experiments
  - TopDownHillClimbing
  - no pruning

- Problems of SECo learners:

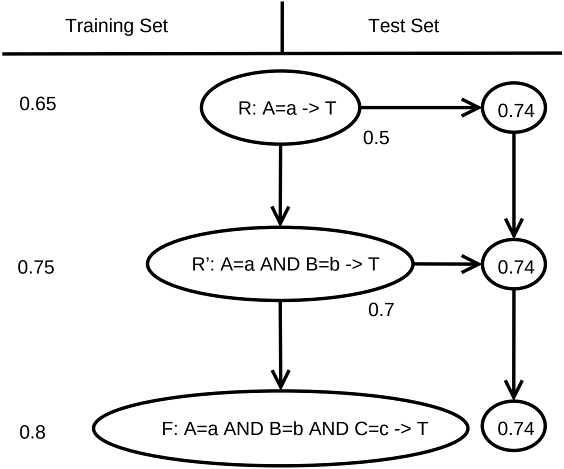
1. Problem of unreliable estimates (measured on the training set)
   - different variances
     - low coverage rules: high variance
     - high coverage rules: low variance
2. Problem of evaluation of candidate rules
   - current heuristics of SECo-algorithms do not differentiate between evaluating a candidate or a final rule

$\Rightarrow$ Search Heuristics merge these two problems

Addressing the problem of unreliable estimates



Addressing the problem of evaluation of candidate rules



Goals

- try to solve the two above-mentioned problems:
  - we try to correct overly optimistic measurements
  - we evaluate candidate rules and final rules differently

find:

- an optimal search heuristic which is learned without a bias towards existing measures and
- two functions that are able to predict true positive/negative coverage values of a rule

Our approach

- to create the meta data let the SeCo algorithm run several times with different settings
- for each run:
  - divide the training set into a training and a test set of equal size (stratified for nominal class values)
  - record statistics of **all** rules on the training set ($P, N, {}^{P}/_{(P+N)}, p, n, {}^{p}/_{P}, {}^{n}/_{N}, {}^{p}/_{(p+n)}, length$)
  - record the positive/negative coverage and the precision of these rules on the test set ($pTest, nTest, {}^{pTest}/_{(pTest+nTest)}$)
- perform a regression on this meta data
- use the resulting function as a heuristic inside the rule learner

The meta data set

Parameters of the meta data generation algorithm:

- 27 UCI datasets with varying characteristics
- 5x2 Cross-Validation (to keep the training and test set of equal size)
- one-against-all class binarization
- 5 standard heuristics employing different biases
  - *precision, accuracy, weighted relative accuracy, laplace, correlation*

Statistics of the data:

- 87,380 examples in total
- ignore rules that cover no example

Regression algorithms and Evaluation methods

Regression algorithms

- a linear regression
  - directly interpretable concept
- neural network (MLP)
  - 1,5,10 (sigmoid) node(s) in the hidden layer, backpropagation run for 1 epoch

Evaluation methods

- Mean Absolute Error

$$MAE(f, \hat{f}) = \frac{1}{m} \sum_{i=0}^{m} |\hat{f}(i) - f(i)|$$

- Main method: macro average accuracy of a 1x10 CV when using the regression model as heuristic

- 27 UCI datasets were used for the meta data generation
- 30 other UCI datasets were used for testing

Predicting Test Set Precision

Coefficients learned by the linear regression

P	N	$P/(P+N)$	p	n	p/P	n/N	$p/(p+n)$	*constant*
0.0001	0.0001	0.7485	-0.0001	-0.0009	0.165	0.0	0.3863	0.0267

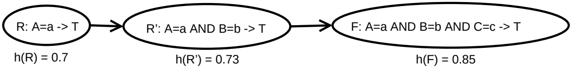
Performance on the 30 ”Test Sets” (macro average accuracy):

- linear regression: 77.43 %
- MLP with 1 node: 77.81 %
- MLP with 5 nodes: 77.37 %
- MLP with 10 nodes: 77.53 %
- *correlation* (for comparison): 77.57 %

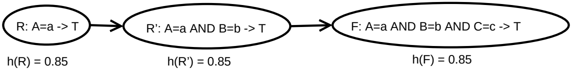
- note that including the length does not increase the accuracy

Final rule vs. immediate rule prediction

- immediate rule prediction: use the actual value of the incomplete rule



- final rule prediction: for all incomplete rules use the value of the final rule they will be refined to



Results (when using final rule prediction as target):

method	avg. accuracy	avg. # conditions
linear regression	77.95 %	95.63
MLP	78.37 %	53.97

Predicting positive/negative coverage

- repeating the experiments with all other heuristics is too expensive
- thus, predict the out-of-sample coverages directly (with the best MLP)

args	*Precision*	Laplace	Accuracy	WRA	Correlation
(p, n)	76.22%	76.89%	75.60%	75.8%	77.57%
$(\hat{p}, \hat{n})$	76.53%	76.80%	75.39%	69.89%	58.09%

- the predictions are not good enough to yield true coverages
  - coverage values that are below 0
  - too optimistic values in regions of low coverage
- only the overfitting problem of precision could be corrected (129.17 vs. 30 conditions in average and a higher accuracy)

Conclusion and further research

- it is possible to learn a heuristic from experience that outperforms standard rule learning heuristics
- it is not that simple to predict true coverage values of rules
- adjust the out-of-range features (P, N, p, n)
- address a third problem in SECo-Rule learning: the problem of local evaluation

The End

- Thank you for your attention!
- Questions?