Adapting the EDSM Algorithm for Ensemble Learning

A Machine Learning Approach to DFA Inference

Author: Radu-Cosmin Dumitru (r.c.dumitru@student.tudelft.nl)

Supervisor: Simon Dieck
Responsible Professor: Sicco Verwer

Introduction

Deterministic Finite Automata (DFAs) are interpretable models often used in classification tasks involving sequential data. Learning DFAs from observed data offers practical benefits, with applications such as software synthesis [1] and protocol analysis [2].

DFA Inference

Given some training data, construct a DFA consistent with that data

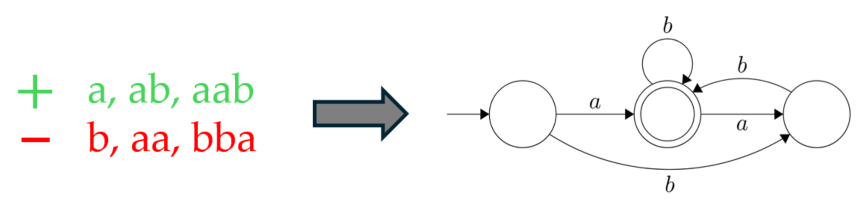

Figure 1: DFA learned from a labeled sample set

Ensemble techniques

Use multiple different models to capture all features in the training data

More robust and generalizable aggregated output

The ensemble is expected to outperform each of its constituent models

Background

Evidence Driven State Merging (EDSM) – competition-winning DFA learning algorithm [3]

Initial Hypothesis

The Prefix Tree Acceptor exactly represents the training set by accepting all positive traces and rejecting all negative ones

The Augmented Prefix Tree Acceptor (APTA) introduces intermediate states that enable merging, thus generalizing beyond the original data

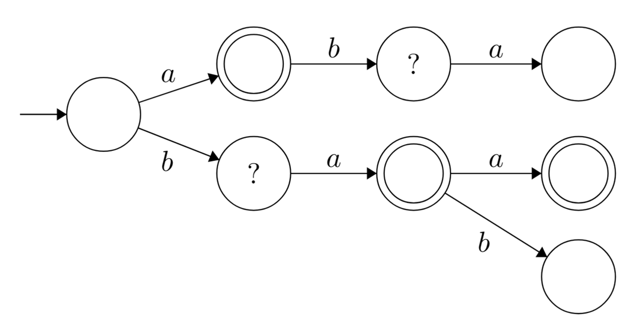

Figure 2: Augmented Prefix Tree Acceptor for $S_+ = \{a, ba, baa\}$, $S_- = \{\epsilon, aba, bab\}$

State Merging

1. Explore states of the DFA in a BFS-like approach
2. Compute the scores of candidate merges based on a heuristic
3. Pick a pair of states with the highest score and merge them
4. Repeat until all states are visited

EDSM Heuristic

A candidate merge is scored based on *evidence* $s = P + N$

P = # positive-positive merges N = # negative-negative merges

performed during the candidate merge

Research Question

How can the EDSM algorithm be adapted to fit within an ensemble learning framework in order to enhance its effectiveness?

Methodology

Ensemble Types

Randomized Models

Introduce noise in the evidence score using a multiplicative random factor $R \sim \mathcal{U}(0, r)$, where r is a hyperparameter of the ensemble

$$s_{new} = (1 - R) \cdot s$$

Boosting

Assign a weight to each state in the initial APTA

Iteratively learn models based on the weighted evidence score

$$s_{new} = \frac{w_a + w_b}{2} \cdot s$$

Validation traces misclassified by new models are simulated on the APTA, and the weight of the final state in each simulated path is increased

Input: S $= \langle S_+, S_- \rangle$: training set, V $= \langle V_+, V_- \rangle$: validation set, n: number of models
Output: Ensemble $\varepsilon = (M_1, M_2, ..., M_n)$

$D \leftarrow$ CreateWeightedAPTA(S)
for i in $1, 2, ..., n$ **: do**
  $M_i \leftarrow$ MergeStates(D)
  $V_i \leftarrow$ Validation traces misclassified by M_i
  $D \leftarrow$ UpdateWeights(D, V_i)
  Add M_i to ε
end for
return ε

Algorithm 1: Boosting process

Ensemble Prediction

An ensemble ε of n models $M_1, M_2, \ldots, M_n$ aggregates their individual predictions $M_i(t)$ through weighted majority voting

$$\text{prediction}(t) = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} w_i \cdot M_i(t) > 0 \\ 0 & \text{otherwise} \end{cases}$$

Diversity Measure

Average Disagreement Rate

The disagreement rate $\Delta_{A,B}$ between two models M_A and M_B is the proportion of traces in the dataset for which their predictions differ

$$\Delta_{A,B} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}_{M_A(t_i) \neq M_B(t_i)}$$

The average of $\Delta_{A,B}$ over all ordered pairs of models M_A and M_B is the diversity measure of the ensemble

Performance Comparison

The ensembles were evaluated against a single DFA learned using the greedy EDSM heuristic

Conclusions

All ensembles achieved significantly better performance on the sparse datasets compared to the baseline EDSM model

Greedy EDSM is more effective on dense data, but the ensembles maintain comparable performance

Although boosting consistently produced the most diverse ensembles, its performance was similar to the randomized models

Results

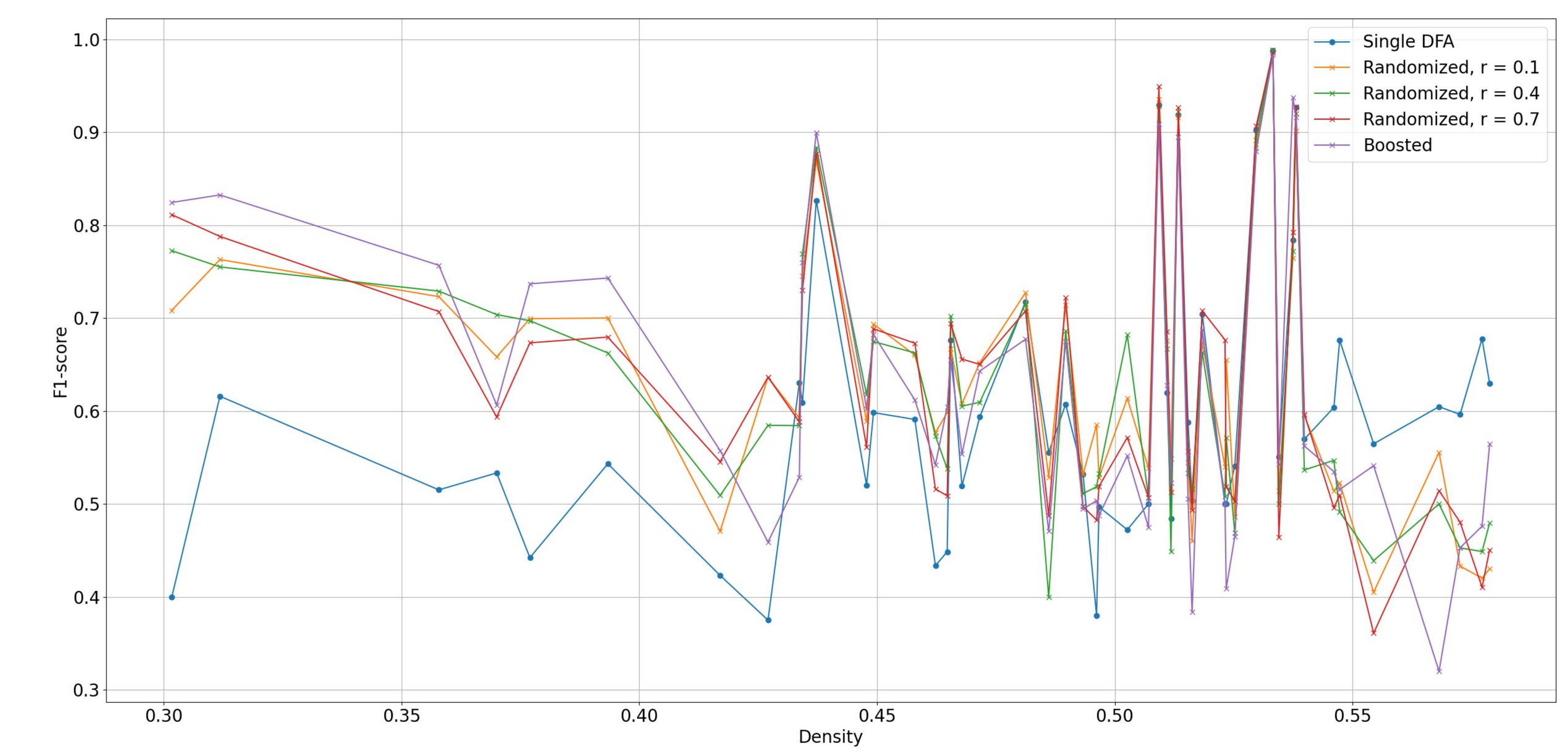

Figure 3: F_1-score of all ensembles and a single DFA, in increasing density of the datasets

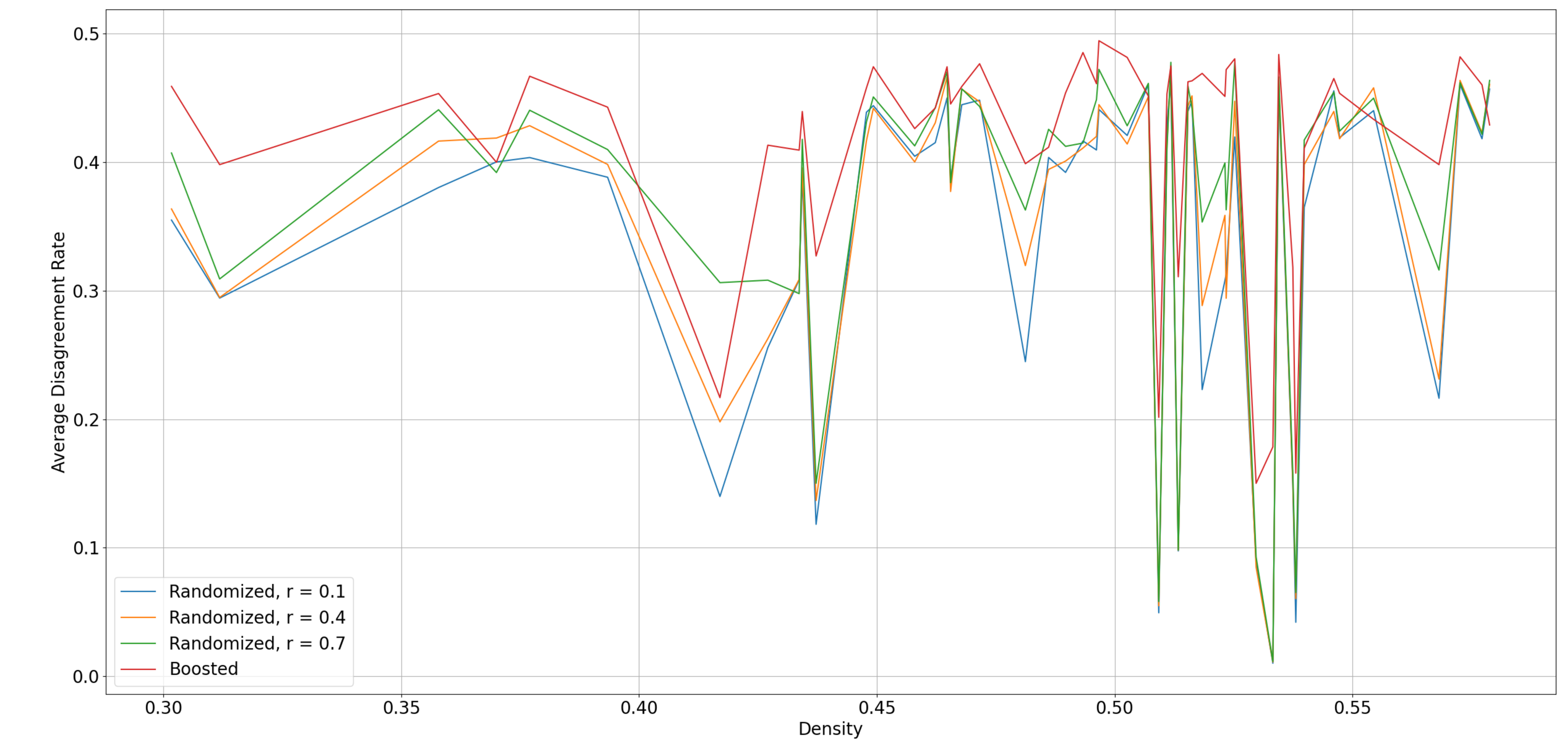

Figure 4: Average Disagreement Rate within each ensemble

Method	Mean	Standard Deviation	Paired t-test vs Single (t, p)
Single	0.5825	0.1351	—
Random, $r = 0.1$	0.5996	0.1399	(1.10, 0.2730)
Random, $r = 0.4$	0.5952	0.1443	(0.81, 0.4203)
Random, $r = 0.7$	0.5966	0.1469	(0.91, 0.3674)
Boosted	0.6032	0.1421	(1.66, 0.1009)

(a) All datasets

Method	Mean	Standard Deviation	Paired t-test vs Single (t, p)
Single	0.5318	0.0967	—
Random, $r = 0.1$	0.6276	0.0807	(6.54, 0.0001)
Random, $r = 0.4$	0.6232	0.0926	(5.68, 0.0001)
Random, $r = 0.7$	0.6226	0.0928	(5.89, 0.0001)
Boosted	0.6214	0.0956	(5.25, 0.0001)

(b) Lower half of datasets

Method	Mean	Standard Deviation	Paired t-test vs Single (t, p)
Single	0.6331	0.1484	—
Random, $r = 0.1$	0.5717	0.1762	(-2.71, 0.0092)
Random, $r = 0.4$	0.5671	0.1774	(-3.01, 0.0041)
Random, $r = 0.7$	0.5706	0.1821	(-2.78, 0.0077)
Boosted	0.5849	0.1750	(-4.00, 0.0002)

(c) Upper half of datasets

Table 1: Statistical analysis of the F_1-score on the entire dataset and on its two density-based halves

References

[1] M.J.H. Heule and S. Verwer. Software model synthesis using satisfiability solvers. *Empirical Software Engineering*, 18(5):825–856, 2013.
[2] S. Wang, F. Sun, H. Zhang, D. Zhan, S. Li, and J. Wang. EDSM-based binary protocol state machine reversing. *Computers, Materials and Continua*, 69(3):3711–3725, 2021.
[3] K. J. Lang, B. A. Pearlmutter, and R. A. Price. Results of the Abbadingo one DFA learning competition and a new evidence-driven state merging algorithm. In *International Colloquium on Grammatical Inference*, pages 1–12. Springer Berlin Heidelberg, 1998.