Adaptive Incremental Learning for Statistical Relational Models Using Gradient-Based Boosting

Yulong Gu and Paolo Missier

Presenter: Yulong Gu

School of Computing, Newcastle University UK

Outline

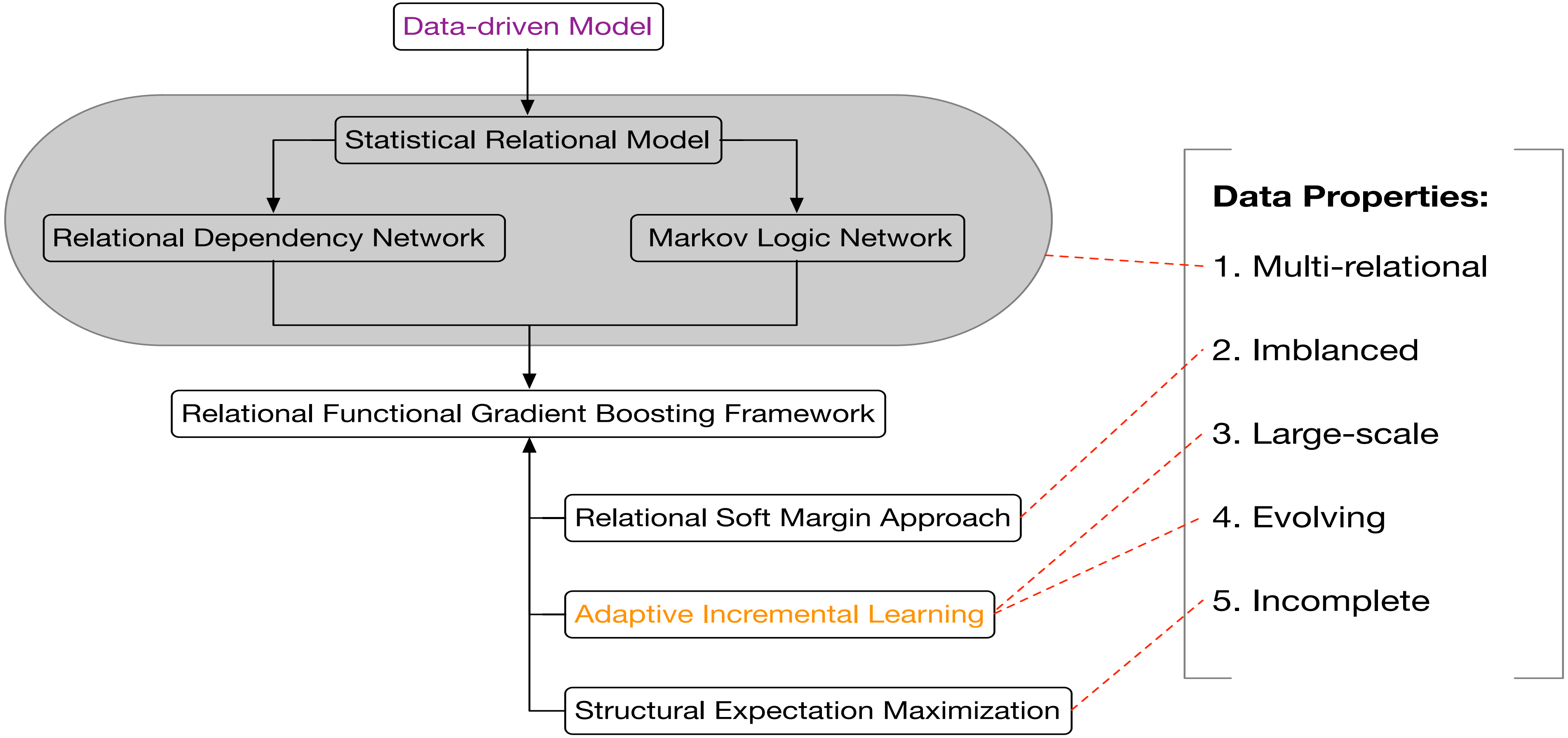
- Background
  - Relational Functional Gradient Boosting (RFGB)
  - Top-down Induction of first-order logical decision trees (TILDE)
  - Concept-Adapting Very Fast Decision Tree (CVFDT)
- Hoeffding Relational Regression Tree (HRRT)
- Rule Stability Metric for CVFDT
- Relational Incremental Boosting (RIB)
- Relational Boosted Forest (RBF)

Problem

Supervised Learning with dataset that:

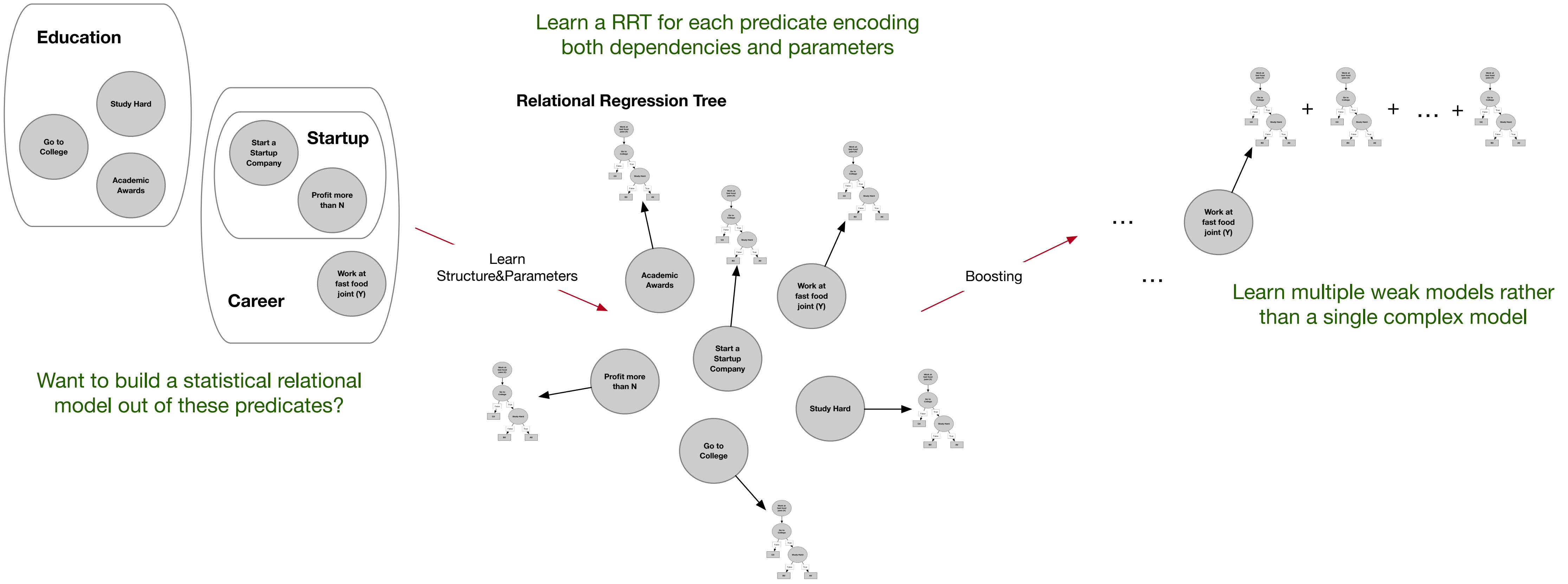
- *Incomplete* – contains missing values
- *Imbalanced* – #negative instances far outnumber #positive instances
- *Large-Scale* – more cost-efficient to update than re-building model
- *Evolving* – concept drifts
- *Multi-relational* – objects are connected in meaningful way

Solution System Design



Relational Functional Gradient Boosting

Learn a RRT for each predicate encoding both dependencies and parameters



Want to build a statistical relational model out of these predicates?

Learn multiple weak models rather than a single complex model

Hoeffding Relational Regression Tree(HRRT)

Incrementally learn Relational Regression Tree?

learn a relational regression tree? ***TILDE***

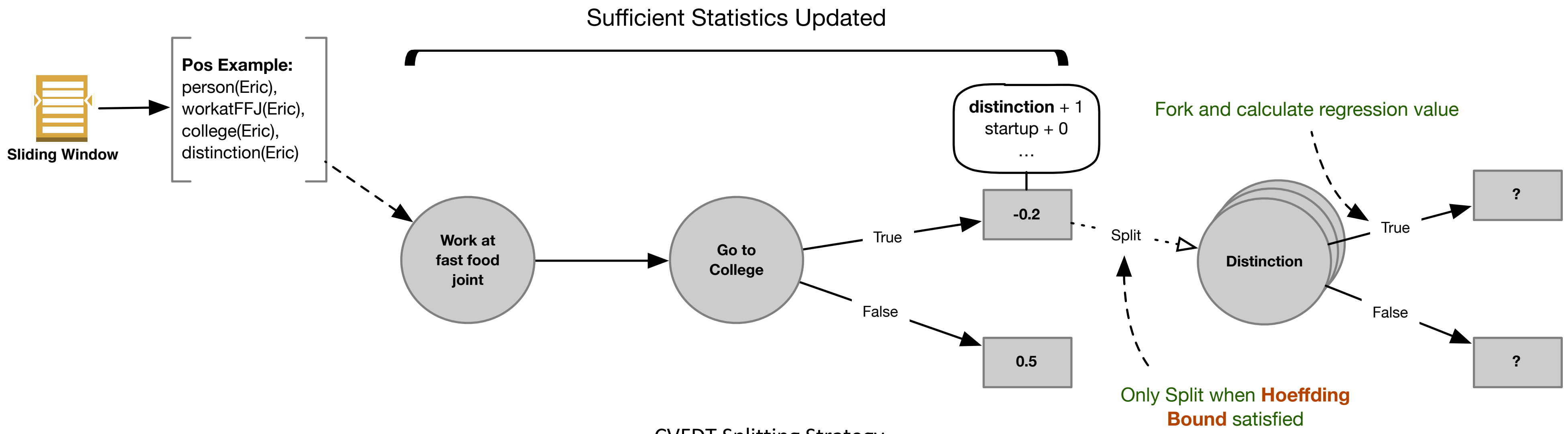
- allows conjunction of predicates
- extensions allow conjunctions of recursive and aggregated predicates

+

Learn regression tree incrementally? ***CVFDT***

- Learn predicate at node with fraction of streaming data
- concept-adapting

= ***HRRT***



CVFDT Splitting Strategy

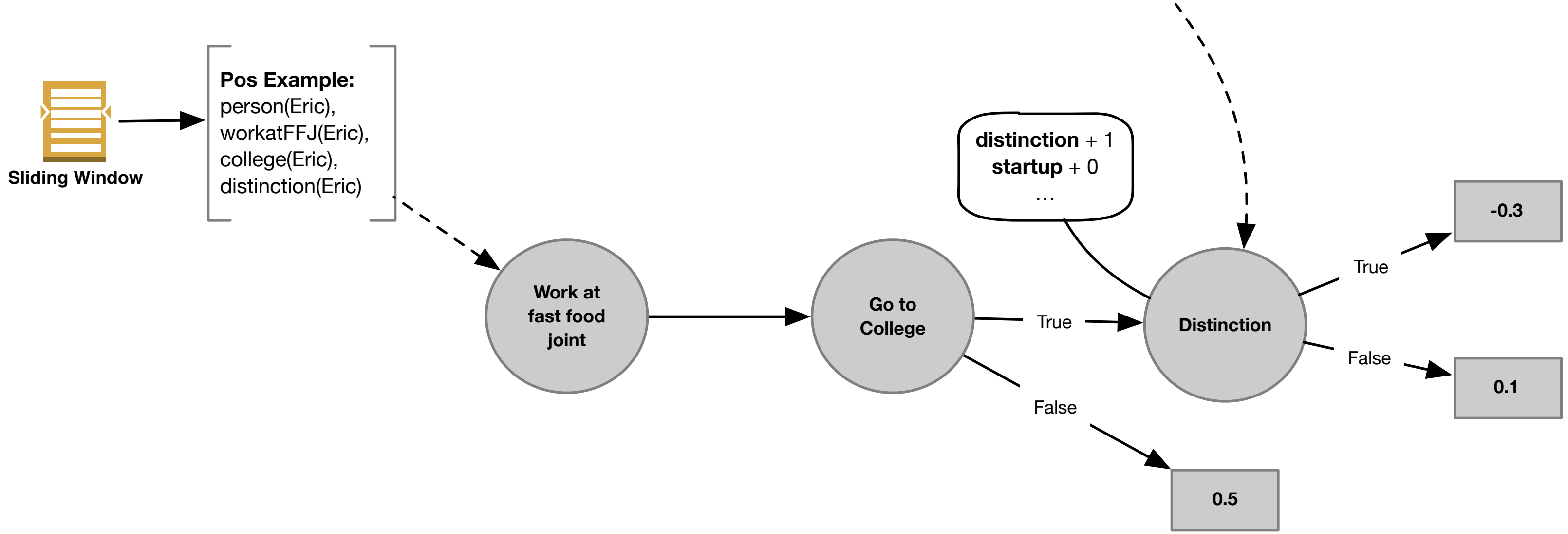
Blockeel, H., & De Raedt, L. (1998). ***TILDE***. *Artificial Intelligence(AI);* Hulten, G.(2001). ***CVFDT***. *KDD*

Hoeffding Relational Regression Tree(HRRT)

Hoeffding Bound:

- With desired confidence, the upper bound of the difference between the true mean and observed mean of a random variable is dependent on the number of observations.

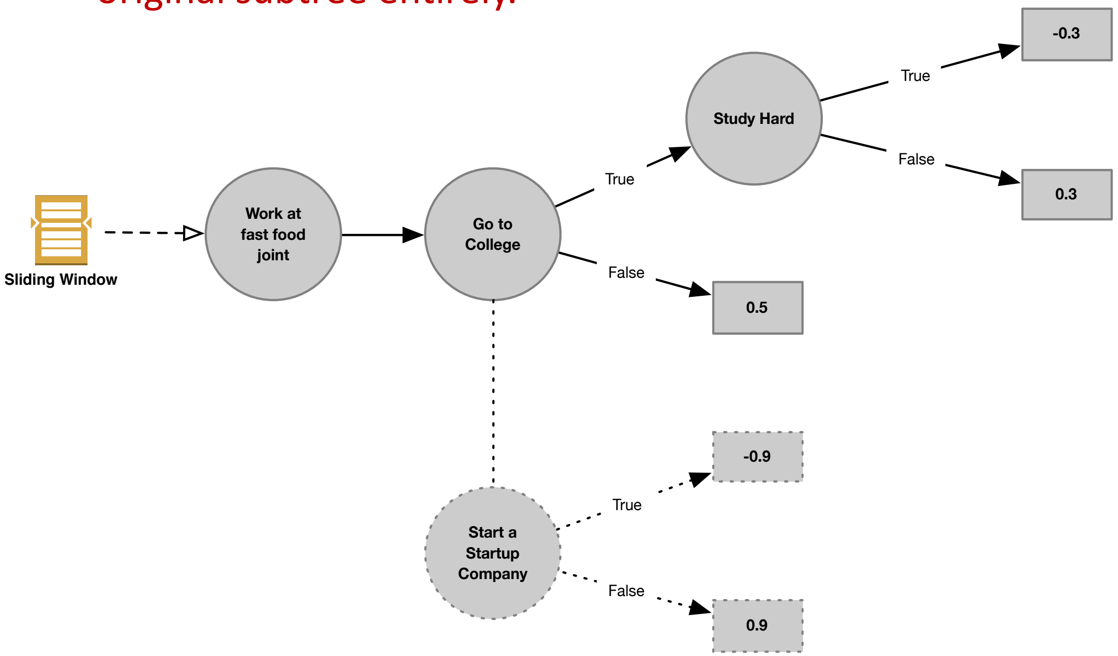
Example:
After update of SS, the node has seen 100 examples, with 99% certainty, the difference between the true $Avg(Eval(Distinction) - Eval(Startup))$ and observed one is less than pre-defined ϵ, HB satisfied, split.



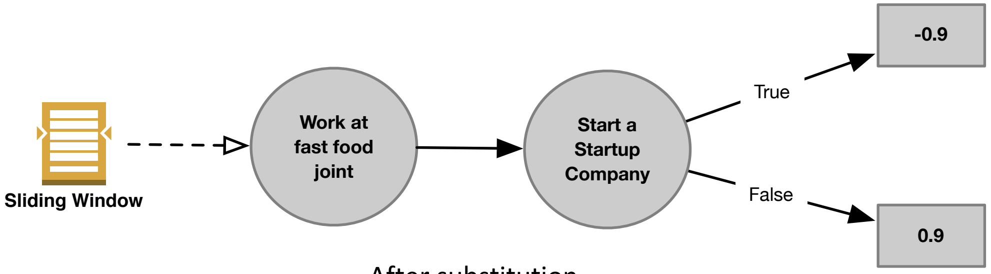
Hoeffding Relational Regression Tree(HRRT)

How does CVFDT adapt to concept drift ?

- Maintain a set of alternative subtrees for each node with different predicates than the original one
- Periodically check HB at each node, if failed, then add new subtree to its subtree set with the best predicate at the moment
- Once one of the subtree outperforms the original one, the wining subtree will replace the original subtree and discard the original subtree entirely.



CVFDT with alternative subtree

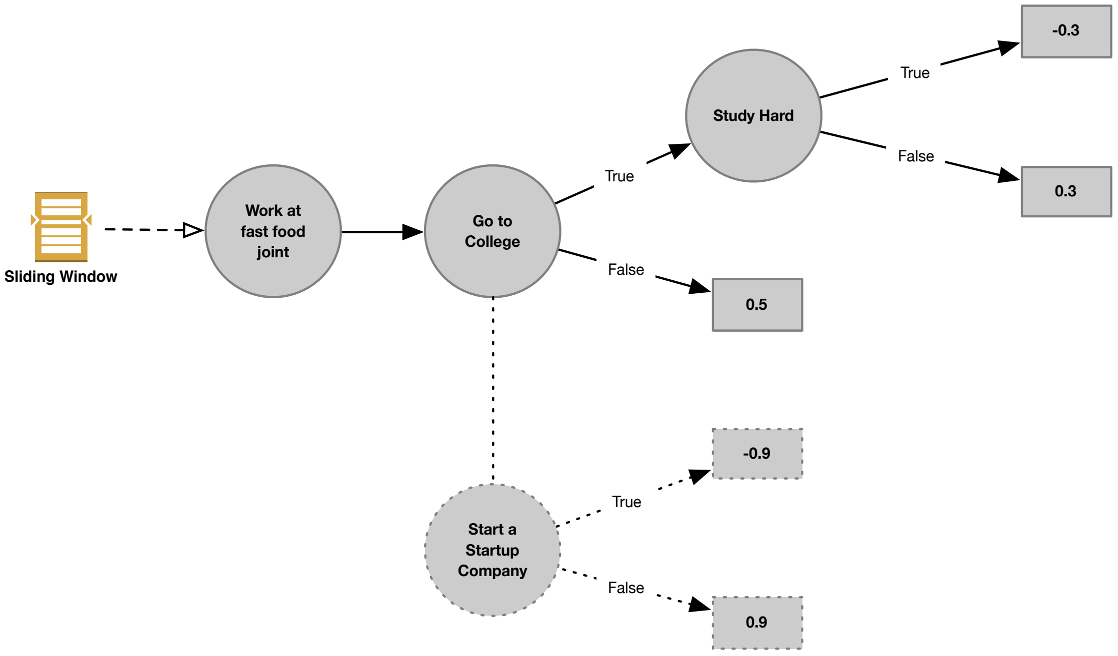


After substitution

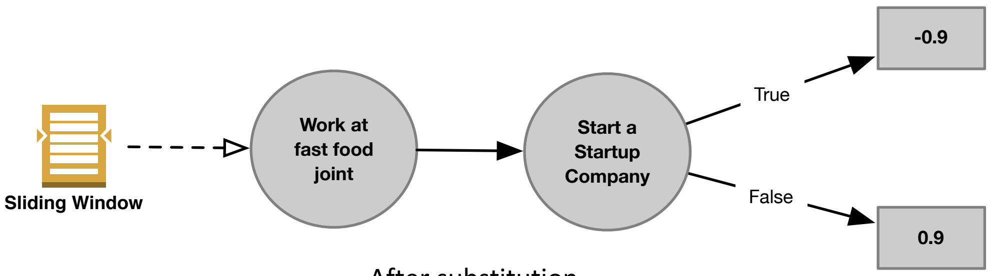
Hoeffding Relational Regression Tree(HRRT)

Why is CVFDT not good enough?

- Less Responsive - new concept will need many counter-examples to invalidate old concept
- larger prediction variance – old concepts are entirely discarded based on relatively small amount of data
- Hard to maintain and analyse – one single complex model



CVFDT with alternative subtree



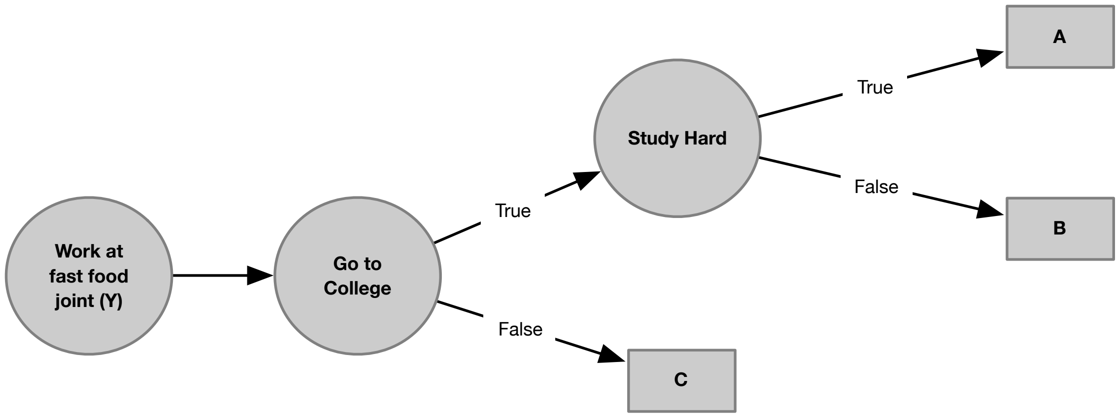
After substitution

Kolter, J.(2007). ***DWM***. J. Mach. Learn. Res.

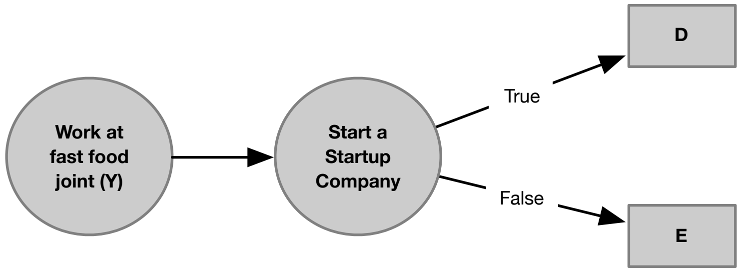
Ensemble Methods for Relational Adaptive Incremental Learning

Ensemble Method for Concept Drift:

- Boosting, Bagging, Weighted Majority...
- Train multiple weak models to represent conflicting rules.
- Each weak model contributes to the final prediction.



Weak Model 1 with weight α



Weak Model 2 with weight β

Boosting:

$$P(Y = True|Pa(Y)) = \alpha A + \beta D, \alpha = \beta = 1$$

Weighted Majority:

$$P(Y = True|Pa(Y)) = \alpha A + \beta D$$

Rule Stability Metric

Definition 1.

- Define the *Rule Stability* of a model as n, the size of the smallest change in sample D that may cause new rule r' to become superior to working rule r. In following equation, D' is D after change:

$$Learner : (Diff(D, D') = n, r) \rightarrow r' \quad (1)$$

When we apply the *Rule stability* to a tree trained with HRRT, we can prove that:

- With confidence $1 - \delta$, the size of the smallest change that may cause r' to become superior to r is:

$$Tolerance = \Delta\bar{G}_{X_a,X_b} - \epsilon \quad (2)$$

- $\Delta\bar{G}_{X_a,X_b}$ is the average of the difference between the scores of test X_a and X_b evaluated by splitting function $G(X_i)$, and ϵ is the parameter obtained from the Hoeffding inequality given n and a desired confidence δ.
- The $Tolerance$ measures the rule stability of an inner node, and we define:

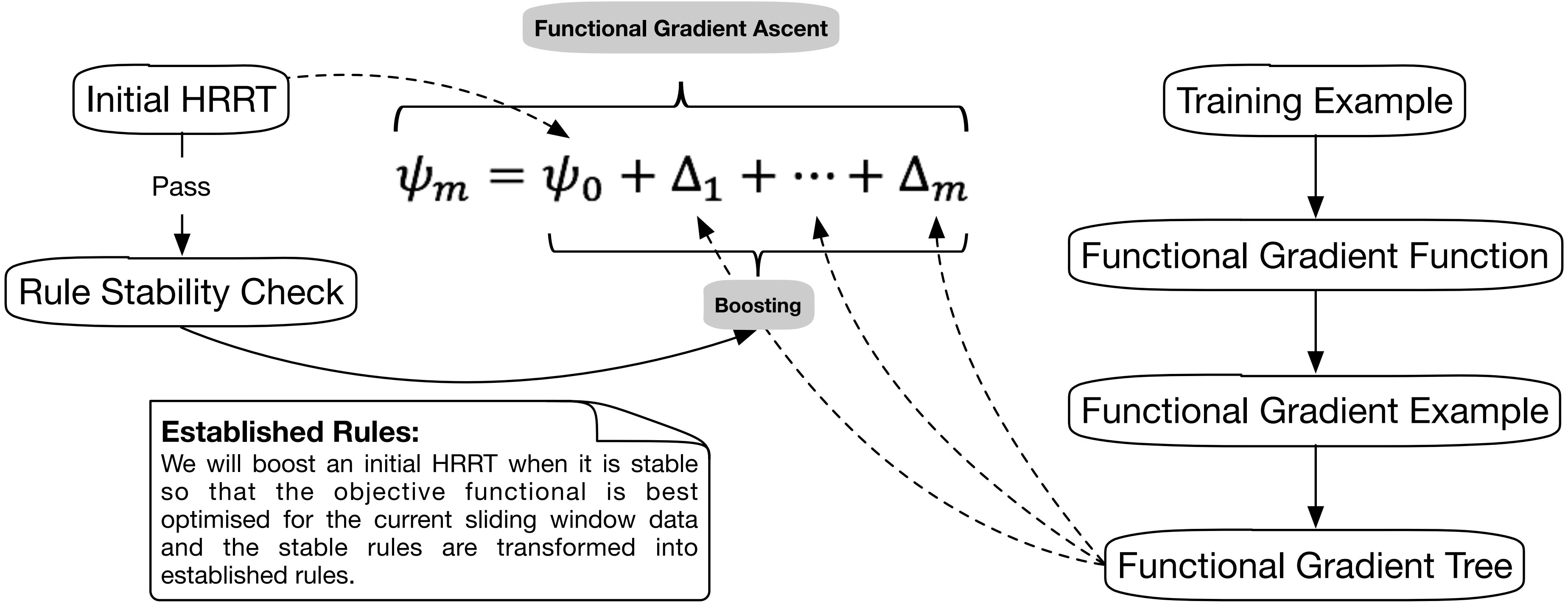
$$TreeTol = \sum_{node=0}^{N} node_{Tolerance} \quad (3)$$

- as the stability of the tree.

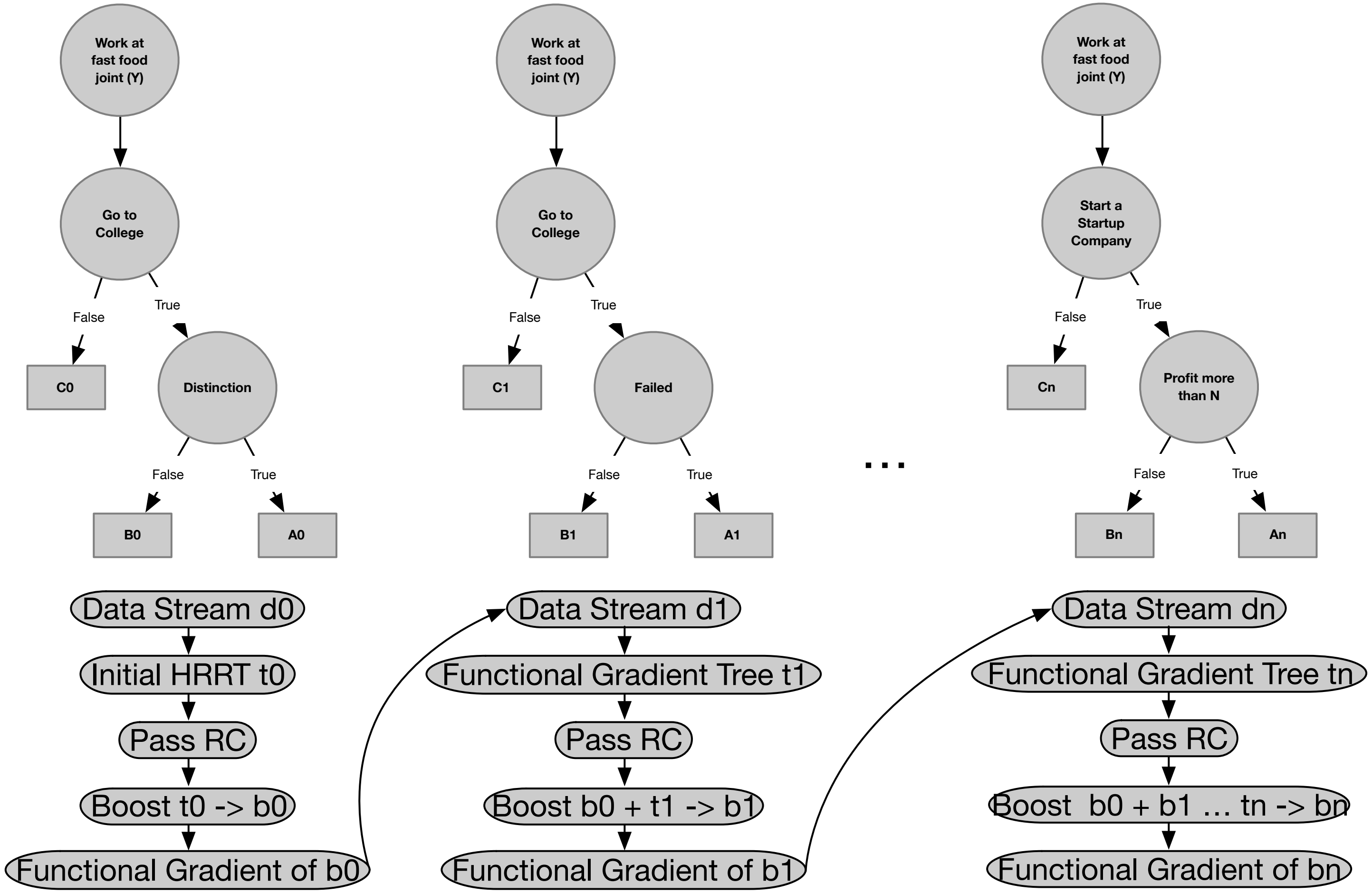
Established Rules

Combine HRRT and Rule Stability to enable Ensemble Methods to handle Concept Drift:

- *When is the weak model good enough to represent current rules?*
  - It passes the rule stability check with current sliding window data and
  - It got boosted using current sliding window data

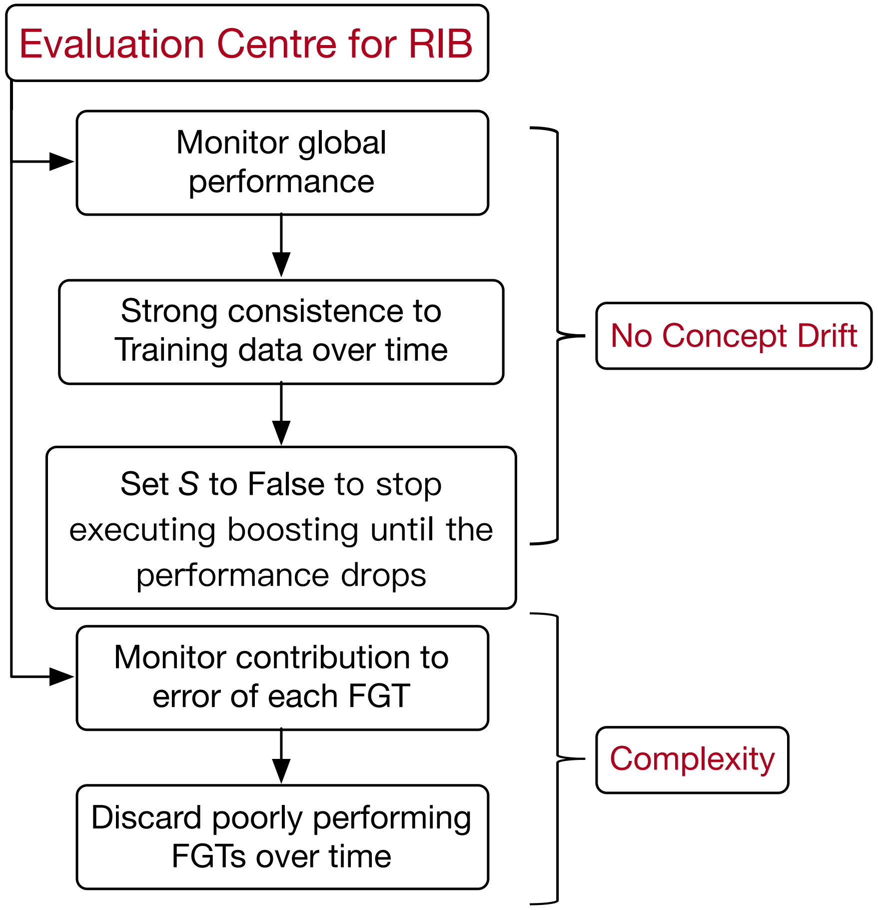


Relational Incremental Boosting



$$P(Y = True|Pa(Y)) = A_0 + A_1 + \cdots + A_n$$

Relational Incremental Boosting

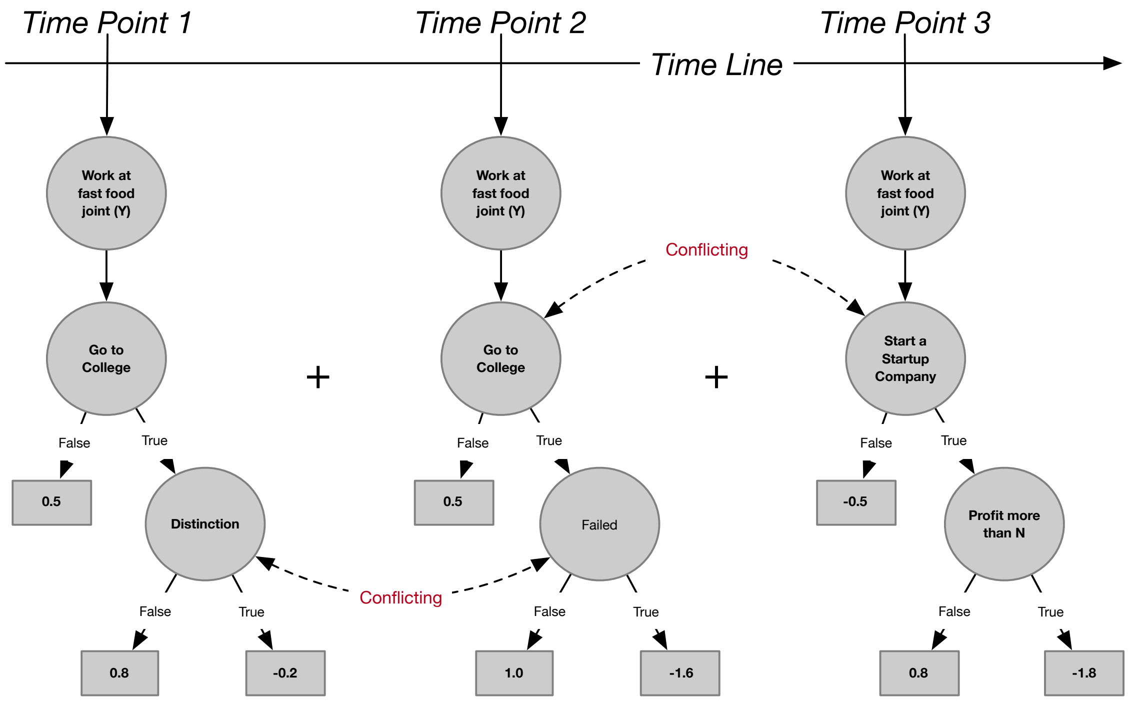


Algorithm 1 Relational Incremental Boosting

procedure RIB($DataStream, p$)
    Initialize empty tree ψ and η
    for each d in $DataStream$ **do**
        After every p examples **do** $\{\psi, S\} \leftarrow EvalCentre(\psi, d)$
        if $\psi.boosted$ **then** $\eta \leftarrow HRRT(\eta, GradExpGen(\psi, d))$
            if $StabilityCheck(\eta)$ **and** S **then** $\psi \leftarrow Boosting(\eta + \psi)$ and reset η
            end if
        else $\psi \leftarrow HRRT(\psi, d)$
            if $StabilityCheck(\psi)$ **then** $\psi \leftarrow Boosting(\psi)$, $\psi.boosted \leftarrow True$
            end if
        end if
    end for
    return $(\psi + \eta)$
end procedure

Relational Incremental Boosting Example

Assume $P(Y|Pa(Y)) = Sig(x)$ is a Sigmoid function, x is the regression value, Y in following examples is predicate 'Work at fast food joint'.



Scenario at *Time Point 1:*

College and Distinction = less likely work at fast food joint in that fast food joint pays less competitive

$P(Y = True|college, distinction) = Sig(-0.2)$

$P(Y = True|college, failed) = Sig(0.8)$

Scenario at *Time Point 2:*

College and Failed = less likely work at fast food joint due to fast food joint pays extremely well over this period

$P(Y = True|college, distinction) = Sig(-0.2 + 1.0 = 0.8)$

$P(Y = True|college, failed) = Sig(0.8 - 1.6 = -0.8)$

Scenario at *Time Point 3:*

Own a Start-up and Profit more than N = less likely work at fast food joint due to tightening job market

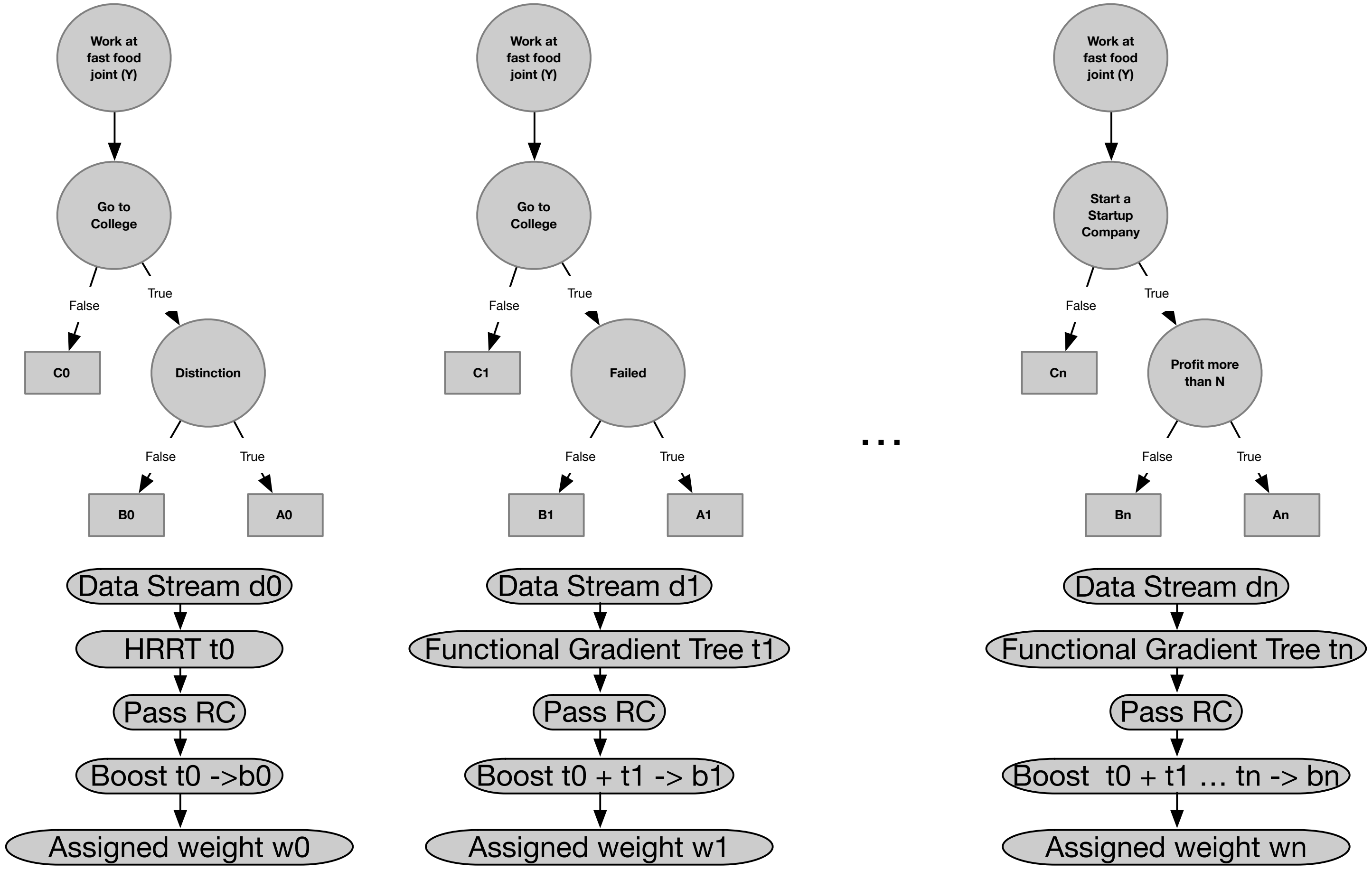
$P(Y = True|college, distinction) = Sig(-0.2 + 1.0 - 0.5 = 0.3)$

$P(Y = True|college, failed) = Sig(0.8 - 1.6 - 0.5 = -1.2)$

$P(Y = True|startup, profitmorethanN) = Sig(0.5 + 0.5 - 1.8 = -0.8)$

The decomposability of ensemble methods allows direct event analysis of time series from the real-time incrementally learned model

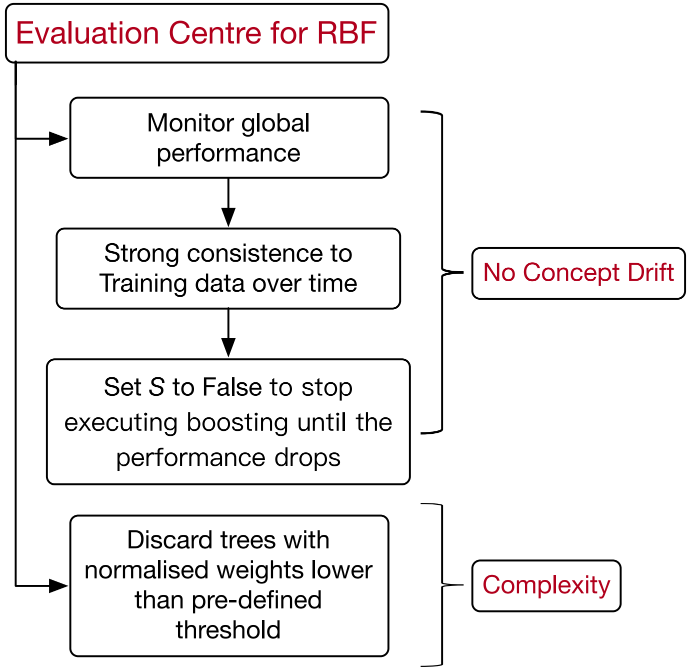
Relational Boosted Forest



$$P(Y = True|Pa(Y)) = \widehat{\boldsymbol{W}} \cdot \boldsymbol{A}, \mathbf{W} = \{w_0, w_1 \dots, w_n\}, \boldsymbol{A} = \{A_0, A_1, \dots, A_n\}$$

Relational Boosted Forest

- The weights update strategy is inspired by Dynamic Weighted Majority(DWM)
- The weights are initialized to 1.
- When a boosted tree makes a mistake in a predictive attempt, the evaluation center will decrease its weight by certain proportion.



Algorithm 2 Relational Boosted Forest

```
procedure RBF(DataStream, p)
    Initialize empty Forest & Weights, empty tree ψ and w ← 1
    for each d in DataStream do
        After every p examples do
            {Forest, Weights, S} ← EvalCentre(Forest, Weights, d)
        ψ ← HRRT(ψ, d)
        if StabilityCheck(ψ) and S then ψ ← Boosting(ψ)
            Add ψ to Forest, w to Weights and reset ψ, w ← 1
        end if
    end for
    return {Forest, Weigths}
end procedure
```

Kolter, J.(2007). ***DWM***. J. Mach. Learn. Res.

Conclusion

- We have proposed three adaptive incremental learning algorithms:
  - Hoeffding Relational Regression Tree (HRRT)
  - Relational Incremental Boosting (RIB)
  - Relational Boosted Forest (RBF)
- All three algorithms can incrementally and adaptively learn the parameters and structure simultaneously for SRL models such as MLNs and RDNs.
- The RIB and RBF extend the classical ensemble methods to relational scenario for handling the concept drifts.
- All three algorithms are compatible with existing RFGB-based algorithms such as Structural EM and Soft Margin Approach. The combination of these extensions allow us to learn a model from a incomplete, imbalanced, large-scale and evolving multi-relational dataset in an incremental manner.

References

- [1] Neville, J., & Jensen, D. (2007). Relational Dependency Networks. *Journal of Machine Learning Research*
- [2] Natarajan, S., Khot, T., Kersting, K., Gutmann, B., & Shavlik, J. (2012). Gradient-based boosting for statistical relational learning: The relational dependency network case. *Machine Learning*
- [3] Yang, S., Khot, T., Kersting, K., Kunapuli, G., Hauser, K., & Natarajan, S. (2015). Learning from Imbalanced Data in Relational Domains: A Soft Margin Approach. *ICDM*
- [4] Heckerman, D., Chickering, D. M., Meek, C., Rounthwaite, R., & Kadie, C. (2000). Dependency networks for inference, collaborative filtering, and data visualization. *The Journal of Machine Learning Research*
- [5] Khot, T., Natarajan, S., Kersting, K., & Shavlik, J. (2015). Gradient-based boosting for statistical relational learning: the Markov logic network and missing data cases. *Machine Learning*
- [6] Blockeel, H., & De Raedt, L. (1998). Top-down induction of first-order logical decision trees. *Artificial Intelligence(AI)*
- [7] Hulten, G., Spencer, L., & Domingos, P. (2001). Mining time-changing data streams. *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD*
- [8] Huynh, T. N., & Mooney, R. J. (2011). Online Structure Learning for Markov Logic Networks. *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, ECML PKDD*
- [9] Domingos, P., & Hulten, G. (2000). Mining high-speed data streams. *KDD*
- [10] Li, R.-H., & Belford, G. G. (2002). Instability of Decision Tree Classification Algorithms. *KDD*
- [11] Kolter, J., Maloof, M.: Dynamic Weighted Majority : An Ensemble Method for Drifting Concepts. J. Mach. Learn. Res. 8, 2755–2790 (2007).