Extended Abstract: An Interpretable Semi-supervised Classifier using Rough Sets for Amended Self-labeling

Isel Grau[1] Dipankar Sengupta[1,2] Maria M. Garcia Lorenzo[3] Ann Nowe[1]

Artificial Intelligence Lab, Vrije Universiteit Brussel, Belgium

Centre for Cancer Research and Cell Biology, Queen's University Belfast, UK

Department of Computer Science, Universidad Central de Las Villas, Cuba

Highlights

- Semi-supervised classifiers combine labeled and unlabeled data during the learning phase in order to improve performance compared to a supervised baseline.
- State-of-the-art semi-supervised classifiers are black boxes.
- We propose an interpretable self-labeling grey-box classifier (SIGb) that uses a black box to estimate the missing class labels and a white box to make the final predictions.
- Rough Set Theory (RST) is used for amending mistakes during the self-labeling.
- Experimental results suggest that the RST amending improves accuracy and interpretability of the self-labeling grey-box, leading to superior results when compared to state-of-the-art semi-supervised classifiers.

Self-labeling Grey-box

The SIGb approach uses a black-box classifier to predict the decision class of the unlabeled data points, while a surrogate white box is used to build an interpretable predictive model based on the whole dataset. The aim is to outperform the baseline supervised white-box alternative, while maintaining a good balance between accuracy and interpretability.

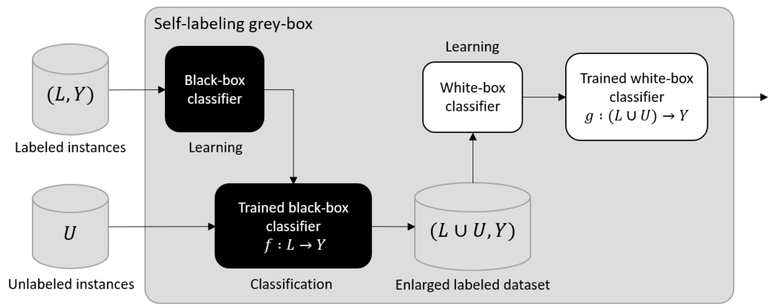


Figure 1. Blueprint of the SIGb architecture. During the first step, labeled data is used for training a black-box model, which assigns labels to the unlabeled data. Later on, a white-box surrogate model is trained on the enlarged dataset, thus resulting in an interpretable model.

However, this solution can propagate misclassifications to the resulting interpretable model since it does not account for two possible sources of class noise:

1. **Class label inconsistency:** very similar data points have different decision classes.
2. **Self-labeling misclassification:** the black-box made an incorrect prediction when labeling the unlabeled data points.

Amending based Rough Set Theory

To address both issues together, we propose to weight the data points after the self-labeling process. Assigning a weight to each data point helps the white box component to focus on learning from the most confident information.

The weight is computed based on Rough Set Theory, a mathematical formalism that allows to handle inconsistency by approximating any set with two other sets: a lower and an upper approximation. These approximations are computed based on the similarity classes of each data point. The goal is to approximate the sets of data points of each decision class.

From the lower and upper approximations of each decision class, three regions of data points are computed:

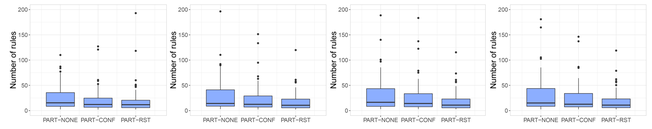
- The positive region: data points that we are sure that are in the decision class.
- The boundary region: data points that might be in the decision class.
- The negative region: data points that we are sure are not in the decision class.

The weight combines the inclusion degree of each data point in the regions of its ground truth label (labeled data) or the class that was assigned during self-labeling (unlabeled data). The following pseudocode summarizes the proposed modification.

Data: Labeled instances (L, Y), Unlabeled instances U
Result: $g : (L \cup U) \rightarrow Y$
begin
  /* Preprocessing: Weight labeled instances */
  forall $(l_j, y_i) \in (L, Y)$ **do**
    $w_{(l_j, y_i)} \longleftarrow |L_{min}|/|L_i|$
  end
  /* Train black-box component with weighted labeled data */
  $f, h \longleftarrow blackboxClassifier.fit(L, Y, w)$
  /* Self-labeling process: Assign a label to unlabeled instances using black-box inference */
  forall $u_k \in U$ **do**
    $y_i \longleftarrow f(u_k)$
    /* Compute weight of instance u_k based on RST inclusion degrees */
    $w_{(x,y_i)} \longleftarrow \varphi\left(\mu^{\mathcal{R}}_{\mathcal{P}(y_i)}(x) + 0.5 * \mu^{\mathcal{R}}_{\mathcal{B}(y_i)}(x) - \mu^{\mathcal{R}}_{\mathcal{N}(y_i)}(x)\right)$
    /* Add the instance to enlarge dataset */
    $(L \cup U, Y) \cup \{(u_k, y_i)\}$
  end
  /* Train white-box component with the weighted $(L \cup U, Y)$ dataset */
  $g \longleftarrow whiteboxClassifier.fit(L \cup U, Y, w)$
  return g
end

Experiments

- 55 benchmark datasets for structured (tabular) semi-supervised classification.
- Four ratios of labeled vs unlabeled instances (10% to 40%).
- Baseline black box: Random Forests
- Baseline white box: Decision lists (PART)



(a) Using 10% of labeled instances. (b) Using 20% of labeled instances. (c) Using 30% of labeled instances. (d) Using 40% of labeled instances.

Figure 2. Number of rules produced by the SIGb without amending, using confidence amending and the RST-based amending. RST-based amending further reduces the number of rules.

Table 1. Mean and standard deviation of kappa coefficient obtained by SIGb and four self-labeling methods from the state-of-the-art. The best performance is highlighted in bold.

	Ratio	10%	20%	30%	40%
SIGb	mean	**0.56**	**0.61**	**0.62**	**0.62**
	(stdev)	**(0.29)**	**(0.27)**	**(0.27)**	**(0.27)**
TT(C45)	mean	0.51	0.55	0.57	0.59
	(stdev)	(0.29)	(0.29)	(0.29)	(0.29)
CB(C45)	mean	0.51	0.55	0.57	0.56
	(stdev)	(0.29)	(0.29)	(0.29)	(0.28)
DCT	mean	0.49	0.54	0.58	0.59
	(stdev)	(0.32)	(0.30)	(0.28)	(0.28)
CT(SMO)	mean	0.48	0.55	0.58	0.60
	(stdev)	(0.31)	(0.29)	(0.29)	(0.29)

Conclusions

- RST-based amending produces more concise sets of rules without affecting the prediction rates by giving more importance to confident instances in the self-labeling.
- SIGb is able to outperform state-of-the-art self-labeling approaches across a standard benchmark of SSC datasets, yet being far more simple in structure than these techniques.