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

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

Isel Grau
Artificial Intelligence Laboratory
Vrije Universiteit Brussel
Brussel, Belgium
igraugar@vub.be

Dipankar Sengupta
Centre for Cancer Research and Cell Biology
Queens University Belfast
Belfast, United Kingdom
D.Sengupta@qub.ac.uk

Maria M. Garcia Lorenzo
Department of Computer Science
Central University of Las Villas
Santa Clara, Cuba
mmgarcia@uclv.edu.cu

Ann Nowe
Artificial Intelligence Laboratory
Vrije Universiteit Brussel
Brussel, Belgium
ann.nowe@vub.be

***Abstract*—Semi-supervised classifiers combine labeled and unlabeled data during the learning phase in order to increase classifier's generalization capability. However, most successful semi-supervised classifiers involve complex ensemble structures and iterative algorithms which make it difficult to explain the outcome, thus behaving like black boxes. Furthermore, during an iterative self-labeling process, mistakes can be propagated if no amending procedure is used. In this paper, we build upon an interpretable self-labeling grey-box classifier that uses a black box to estimate the missing class labels and a white box to make the final predictions. We propose a Rough Set based approach for amending the self-labeling process. We compare its performance to the vanilla version of our self-labeling grey-box and the use of a confidence-based amending. In addition, we introduce some measures to quantify the interpretability of our model. The experimental results suggest that the proposed amending improves accuracy and interpretability of the self-labeling grey-box, thus leading to superior results when compared to state-of-the-art semi-supervised classifiers.**

***Index Terms*—explainable artificial intelligence, grey-box model, rough sets, semi-supervised classification, self-labeling**

I. Introduction

Gathering data examples for training a machine learning classifier in a real-world scenario is often simple, but the process of assigning labels to the examples can be costly in terms of money, time or effort. In such scenarios we might obtain datasets with more unlabeled than labeled data. Semi-supervised classification (SSC) overcomes this issue using both labeled and unlabeled data for training a classifier. The goal is to increase the generalization ability of the classifier when compared to another that only uses the labeled data.

In such scenarios, SSC algorithms are useful under the assumption that the unlabeled data contains information which is relevant for prediction, hence leading to the desired performance improvement. Therefore, unlabeled data should follow the distribution of the labeled data in order to predict the posterior distribution correctly. SSC families of methods involve some drawbacks coming from the variety of assumptions they make for their workings. For example, self-labeling classifiers rely on the prediction of one or more base classifiers to repeatedly increase the size of the labeled dataset by predicting the unlabeled instances. Although self-labeling approaches such as Co-training [1], Self-training [2], Pseudo-label [3] and their variants perform quite well in terms of accuracy, they often result in complex structures failing to give insight into their decision process. When using simpler strategies, e.g. self-training by adding the instances incrementally, errors can be easily propagated. The use of amending procedures [4] allows selecting or weighting the self-labeled instances for enlarging the labeled dataset and improve overall performance. However, designing effective amending strategies is still an open problem.

An increasing requirement observed in machine learning is to obtain not only precise models but also interpretable ones. End users often demand an insight into how an algorithm arrives at a particular outcome and need an explanation of the decisions to some extent. Although some studies [5]–[7] attempt to formalize terms such as interpretability or explainability, a common conclusion is that a certain grade of global interpretability can be reached through the use of more transparent techniques as proxies for solving a task. In this paper, we refer to such models (e.g., linear regression, decision trees or rule induction algorithms) as white boxes, as opposed to the less interpretable black-box ones (e.g. artificial neural networks or support vector machines). Black boxes are normally more accurate techniques that learn exclusively from data but they are not easily understandable at a global level. On the other hand, white boxes refer to models which are constructed based on laws or principles of the problem domain, or those who are built from data but their structure allows

This work was supported by the IMAGica project, financed by the Interdisciplinary Research Programs and Platforms (IRP) funds of the Vrije Universiteit Brussel; and the BRIGHTanalysis project, funded by the European Regional Development Fund (ERDF) and the Brussels-Capital Region as part of the 2014-2020 operational program through the F11-08 project ICITY-RDI.BRU (icity.brussels).

for explanations or interpretation, since pure white boxes rarely exists. White boxes lead to intrinsically interpretable models [8], while post-hoc methods for interpretability such as LIME [9] or SHAP [10] aim to generate explanations preserving the black box's accuracy. Another alternative to reach interpretability is using white boxes as global surrogates [8] for distilling previously trained black boxes. While the white boxes attempts to explain the problem domain directly, the latter is devoted to explain the domain by approximating the predictions produced by a black-box classifier.

In this paper, we study the SSC problem from the interpretability angle. We build upon a simple yet effective semi-supervised classifier termed *self-labeling grey-box* (SlGb) [11], [12] that exploits the strength of black-box models being good classifiers with the interpretability of white boxes. We focus our study in the use of rule-based white boxes, since they are a clear proxy for interpretability both in a global and local perspective. As a first contribution of the paper, we introduce a Rough Set Theory (RST) based strategy to reduce the effect of misclassifications when building the enlarged dataset. By weighting instances based on their inclusion to rough sets regions of each decision class, we expect to generate more compact rule sets in the learning process of the white box. As a second contribution, we introduce some measures to asses the interpretability of the resulting grey-box model. These measures can be extended to other rule based interpretable models. Experimental results using 55 benchmark datasets show that our SlGb outperforms other state-of-the-art semi-supervised classifiers in terms of accuracy. Moreover, the proposed RST-based amending improves interpretability by reducing the number of rules needed for achieving a good performance.

The rest of this paper is structured as follows. Section II formalizes the semi-supervised classification problem and introduces the theoretical background on RST. Section III describes the SlGb approach and proposes the RST-based amending of the self-labeling performed by the black-box classifier. Section IV discusses the simulation results, which cover both the performance and interpretability angles. Section V formalizes the concluding remarks.

II. Theoretical Background

In this section, we introduce the theoretical background supporting our contribution, namely: semi-supervised classification and rough sets theory.

A. Semi-supervised Classification

Supervised classification is about identifying the right category (among those in a predefined set) to which an observation belongs. These observations (henceforth called instances) are often described by a set of numerical and/or nominal attributes. Solving this problem implies to define a mapping $f : X \rightarrow Y$ that assigns to each instance $x \in X$, described by a set of attributes $A = \{a_1 \ldots, a_p\}$, a decision class $y \in Y$. The mapping is learned from data in a supervised fashion, i.e., by relying on a set of previously labeled examples.

Semi-supervised learning techniques attempt to use both labeled and unlabeled instances during the learning process for increasing the prediction capacity when only labeled data is used. More formally, in an SSC scenario we have a set of m instances $L = \{l_1, \ldots, l_m\}$ which are associated with their respective class labels in Y, and a set of n unlabeled instances $U = \{u_1, \ldots, u_n\}$, where usually $n > m$. Overall, the performance of SSC models can be evaluated as follows: (1) transductive learning, which only attempts to predict the labels for the given unlabeled instances in U; or (2) inductive learning, which tries to infer a mapping $g : L \cup U \rightarrow Y$ for predicting the class label of any instance. For this study we focus on inductive learning.

The SSC literature reports several techniques including transductive support vector machines [13], graph-based methods [14], generative mixture models [15], self-labeling techniques [16] or semi-supervised generative adversarial networks [17]. In particular, self-labeling refers to a wide family of versatile semi-supervised methods that employ one or more base classifiers for enlarging the available labeled dataset by assuming the predictions they produce on the unlabeled data are correct. Within this family, self-training approaches [2] are wrapper classifiers, which rely on the prediction of only one base classifier to repeatedly increase the size of the labeled dataset by predicting the unlabeled instances. The instances are added incrementally, in batch [18] or in an amending procedure [4]. The use of amending procedures allows selecting or weighting the self-labeled instances for enlarging the labeled dataset, hence avoiding error propagation. A wide experiment conducted in [16] shows that CoTraining using support vector machines as a base classifier [1], TriTraining using C4.5 decision tree [19], CoBagging using C4.5 [1] and Democratic Co-learning (as an ensemble of naïve Bayes, C4.5 and k-nearest neighbors) [20], are the best performing self-labeling classifiers evaluated against a comprehensive collection of benchmark datasets. A full review on semi-supervised classification techniques is out of the scope of this paper but the reader can refer to [21] for a wide and recent survey on this field.

B. Rough Set Theory

Rough set theory (RST) [22] allows handling uncertainty in the form of class inconsistency in real-world applications. Given a decision system $DS = (\mathcal{U}, A \cup \{d\})$ where the universe of instances $\mathcal{U}$ is described by a non-empty finite set of attributes A and its respective decision class d, any concept (subset of instances) $X \in \mathcal{U}$ can be approximated by two crisp sets. These sets are called lower and upper approximations of X ($\underline{B}X$ and $\overline{B}X$, respectively) and can be computed taking into account an equivalence relation, as follows:

$$\underline{B}X = \{x \in \mathcal{U} \mid [x]_B \subseteq X\} \tag{1}$$

$$\overline{B}X = \{x \in \mathcal{U} \mid [x]_B \cap X \neq \emptyset\} \tag{2}$$

The equivalence class $[x]_B$ gathers the instances in the universe $\mathcal{U}$ which are inseparable according to a subset of attributes $B \subseteq A$. From the formulations of upper and lower approximation, we can derive the positive, negative and boundary regions of any subset $X \in \mathcal{U}$. The positive region $\mathcal{P}(X) = \underline{B}X$ includes those instances that are surely contained in X; the negative region $\mathcal{N}(X) = \mathcal{U} - \overline{B}X$ denotes those instances that are surely not contained in X, while the boundary region $\mathcal{B}(X) = \overline{B}X - \underline{B}X$ captures the instances whose membership to the set X is uncertain, i.e., they might be members of X.

The classic RST is regularly defined over a subset of discrete attributes, thus generating a partition of $\mathcal{U}$. A more relaxed formulation of RST establishes the inseparability between instances based on a weak binary relation. Equation (3) formalizes the similarity relation used in this paper,

$$\mathcal{R} : x_i \mathcal{R} x_j \rightarrow \delta(x_i, x_j) \geq \varepsilon \tag{3}$$

where $\delta(x_i, x_j)$ computes the extent to which x_i and x_j are deemed inseparable as indicated by the similarity threshold ε. Under this assumption, the universe is arranged in similarity classes that are not longer disjoint but overlapped. In this paper, we use $\varepsilon = 0.98$ and the *Heterogeneous Euclidean-Overlap Metric* [23] to measure the inseparability degree between two instances. It is worth mentioning that other configurations are also possible.

Once the covering of the decision space is generated according to the similarity function, several RST-based measures can be computed for measuring the uncertainty contained in a dataset [24]. In the following section, we adopt one of these measures to weight the instances belonging to the enlarged training set obtained after performing the self-labeling process.

III. Amended Self-labeling Grey-box

In this section, we build upon the SlGb method by proposing an amending procedure using RST. The RST measures determine the weight of the instances in the self-labeling process based on the uncertainty in the form of class label inconsistency lying within the enlarged dataset.

A. Self-labeling Grey-box Approach

The SlGb approach [11] uses a black-box classifier to predict the decision class of the unlabeled instances, while a surrogate white box is used to build an interpretable predictive model based on the whole instance set. The aim is to outperform the base white-box component using only the originally labeled data, while maintaining a good balance between performance and interpretability.

The SlGb learning process (see Figure 1) is performed in a sequential order. In a first step, we provide the available labeled dataset (L, Y) to a black-box classifier for training. Once the supervised learning is completed, the black-box component has learned a function $f : L \rightarrow Y$, where $f \in F$, being F the hypothesis space that associates each instance with a class label. The f function can be computed from the scoring function $h : L \times Y \rightarrow [0, 1]$ such that $f(x) = argmax_{y \in Y}\{h(l, y)\}, l \in L$. Thereafter, the trained black-box component is used for generating new tuples (u, y) by mapping all unlabeled instances $u \in U$ to a class label $y \in Y$ as $y = f(u)$, adding a self-labeling character to the approach. From this step we obtain an enlarged training set $(L \cup U, Y)$ comprising the original labeled instances and the extra labeled ones.

In the second step, the enlarged training set $(L \cup U, Y)$ is used to train a surrogate white-box classifier. Once the learning process in the white-box component is completed, we obtain a function $g : (L \cup U) \rightarrow Y$ resulting in a classifier which is more likely to have better generalization capabilities than the original white-box component, when trained on only the labeled data.

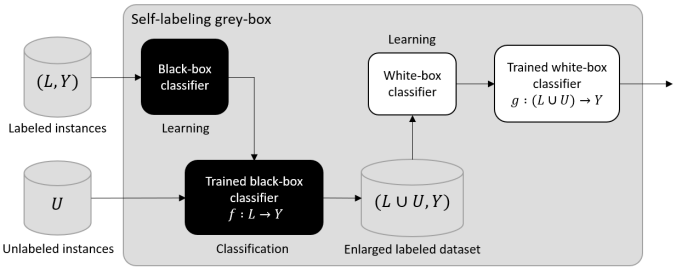


Fig. 1: Blueprint of the SlGb architecture. During the first step, labeled data is used for training a black-box model, which assigns labels to the unlabeled data. Later on, a white-box surrogate model is trained on the enlarged dataset, thus resulting in an interpretable model.

When applying self-labeling, we should be aware of the risk of having imbalanced data with respect to the class labels. It might be easier to obtain unlabeled data of a certain class, for example, in the context of credit fraud detection or rare diseases classification. In order to deal with this problem, our approach additionally incorporates a simple strategy for balancing instances as a preprocessing step. This weight is computed as follows:

$$w_{(l_j, y_i)} = |L_{[y_{min}]}| / |L_{[y_i]}| \tag{4}$$

where $L_{[y_i]}, L_{[y_{min}]} \subset L$ denote the sets of labeled instances that are mapped to the class label y_i and the minority class y_{min}, respectively. In this way we assign higher importance to instances belonging to the minority class.

In general, the SlGb approach is only based on the general assumption of SSC methods: the distribution of unlabeled instances helps elucidate the distribution of all examples. In addition, our approach allows retaining the inherent interpretability of the chosen white-box surrogate, which will be assessed later in section IV.

B. Rough Set Amending

As mentioned, the motivation for using of amending strategies is based on the fact that the black box could produce wrong labels for unlabeled instances, which can be propagated

in the self-labeling. A confidence based amending procedure proposed in our previous work [11] defines a weight for the unlabeled instances relying on the confidence of the classification of the black-box component. The weight $w_{(u_k,y_i)}$ is computed as the value of the scoring function of the black-box base classifier $h(u_k, y_i)$, which expresses the calibrated probability of u_k being correctly assigned to the y_i class.

However, there is no guarantee that the knowledge concerning the original labeled instances does not contain uncertainty in the form of class label inconsistency in the classification. To address both situations together, we propose a mechanism to weight the instances after the self-labeling process. Unlike the confidence-based amending [11], this amending procedure is adopted for the entire enlarged dataset, instead of only the self-labeled instances.

More explicitly, our proposal is based on the inclusion degree of both labeled and self-labeled instances into the RST granules. Let $X = L \cup U$ and $d = y$. Let $\mu^{\mathcal{R}}_{\mathcal{P}_{(y_i)}}(x)$, $\mu^{\mathcal{R}}_{\mathcal{B}_{(y_i)}}(x)$ and $\mu^{\mathcal{R}}_{\mathcal{N}_{(y_i)}}(x)$ be the membership degrees of any instance x to the positive, boundary and negative region of each class label y_i, respectively. These membership degrees are computed from the inclusion degree of the similarity class of x into each information granule,

$$\mu^{\mathcal{R}}_{\mathcal{P}_{(y_i)}}(x) = \frac{|\bar{\mathcal{R}}(x) \cap \mathcal{P}(X_{[y_i]})|}{|\mathcal{P}(X_{[y_i]})|} \tag{5}$$

$$\mu^{\mathcal{R}}_{\mathcal{B}(y_i)}(x) = \frac{|\bar{\mathcal{R}}(x) \cap \mathcal{B}(X_{[y_i]})|}{|\mathcal{B}(X_{[y_i]})|} \tag{6}$$

$$\mu^{\mathcal{R}}_{\mathcal{N}(y_i)}(x) = \frac{|\bar{\mathcal{R}}(x) \cap \mathcal{N}(X_{[y_i]})|}{|\mathcal{N}(X_{[y_i]})|} \tag{7}$$

where $\bar{\mathcal{R}}(x)$ is the similarity class associated with the instance x, whereas $X_{[y_i]}$ denotes the set of instances with label y_i.

Equation (8) computes the weight for the instance x belonging to the enlarged dataset, given its label y_i and a similarity relation $\mathcal{R}$. The sigmoid function $\varphi(x) = 1/(1+e^{-x})$ is used to maintain the weight in the $(0, 1)$ range.

$$w_{(x,y_i)} = \varphi\left(\mu^{\mathcal{R}}_{\mathcal{P}(y_i)}(x) + 0.5 * \mu^{\mathcal{R}}_{\mathcal{B}(y_i)}(x) - \mu^{\mathcal{R}}_{\mathcal{N}(y_i)}(x)\right) \tag{8}$$

Observe that the boundary information is also interesting, since a high inclusion degree of an instance in the boundary region of a class is to some extent a positive evidence. This knowledge can be reinforced or diluted according to the evidence coming from the inclusion degrees in the other two regions. We expect that including this type of information in the learning process of the white box leads to more compact rule sets while obtaining comparable or improved performance. It is important to note that the amending process is only carried out in the learning phase of the self-labeling grey-box. Therefore, the amending strategies do not affect the transparency of the white-box surrogate during the inference on new cases.

Data: Labeled instances (L, Y), Unlabeled instances U
Result: $g : (L \cup U) \rightarrow Y$
begin
  `/* Preprocessing: Weight labeled instances according to Eq. (4) */`
  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 `according to Eq.(8) */`
    $w_{(u_k,y_i)} \longleftarrow h(u_k, y_i)$
    `/* 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

Algorithm 1: Self-labeling grey-box learning algorithm with rough sets based amending.

The pseudocode in Algorithm 1 formalizes the SlGb approach incorporating this step. Overall, the proposed rough set amending comprises an alternative to the use of incremental or batch procedures, thus reducing the computational burden of the self-labeling process.

IV. Experiments and Discussion

In this section, we evaluate the proposed RST amending against the vanilla version of SlGb [12] and the confidence based amending proposed in our previous works [11]. The experimental design includes 55 benchmark datasets available on KEEL repository [25] with partitions for 10-fold cross validation. Four ratios of labeled instances in the training set (from 10% to 40%) allow studying the influence of the amount of labeled examples in the overall performance. These datasets comprise different characteristics: the number of instances ranges from 100 to 19000, the number of attributes from 2 to 90, and the number of decision classes from 2 to 28. Moreover, we have 25 datasets with different degrees of class imbalance and roughly half of the datasets are multiclass problems[1].

[1]Code, datasets and results for individual datasets using different measures are provided for reproducibility purposes at gitlab.ai.vub.ac.be/igraugar/slgb_scripts/tree/paper

There are several algorithms that can be adopted as base classifiers for the SlGb approach. On the one hand, the selected classifier for the base black box should exhibit a strong predictive capability as it is used to determine the decision class of unlabeled instances. On the other hand, for the white-box component any classifier that can act as a surrogate model for interpretability can be used. On [26] we report on a extended experimental study supporting the choice of random forests and PART decision lists as best performing black box and white box respectively, from a pool of 9 combinations of base classifiers. Hereinafter, the SlGb will refer to this combination of base classifiers. The hyperparameters used in the experiments are specified below:

- Random forests (RF) [27]: Bagging of random trees composed of decision trees without pruning considering m randomly chosen attributes at each node. Hyperparameters: 100 trees, minimum number of objects per leaf: 2, number of random attributes: $\log_2(attributes) + 1$.
- PART decision list (PART) [28]: Decision list using separate-and-conquer for building rules. Generates a partial C4.5 decision tree in each iteration and makes the best leaf into a rule. Hyperparameters: minimum number of objects per leaf: 2, confidence factor for pruning: 0.25, uses subtree raising operation when pruning.

In order to measure the configurations in terms of prediction rates we report the Cohen's kappa coefficient [29]. This measure estimate the inter-rater agreement for categorical items and ranges in $[-1, 1]$, where -1 indicates no agreement between the prediction and the actual values, 0 means no learning (i.e., random prediction), and 1 total agreement or perfect performance. While accuracy is considered mainstream when measuring classification rates, the kappa is a more robust measure since this coefficient takes into account the agreement occurring by chance, which is especially relevant for datasets with class imbalance.

Unlike other experiments reported in the literature, the one developed in this section evaluates both algorithms' performance and interpretability, when having different percentages of labeled instances. In the next subsection, we propose new evaluation measures that go beyond the prediction rates.

A. Interpretability Measures

Although obtaining good predictions is pivotal for any classification model, our research is also concerned with the interpretability issues. Toward exploring results further, we propose two new measures to evaluate models' interpretability via a quantifiable proxy. The first measure can be used in the context of self-labeling and the second one is applicable to any model containing explanation units.

According to [6], there are three main forms of evaluating interpretability: application-grounded, human-grounded and functionally-grounded metrics. The functionally-grounded approach is the only one not requiring human experiments and collaboration. As an alternative it uses desiderata for interpretability (e.g. transparency, trust, etc.) as a proxy for assessing the quality of the model. Although this form of evaluation is the most commonly found in literature, the proposed measures are predominantly related to the context of fuzzy rule-based systems [30].

Since we are working with benchmark datasets, we use the functionally-grounded approach for creating measures based on the simplicity as a mean for gaining transparency and simulatability (i.e. a human is able to simulate and reason about the model's entire decision-making process). The first measure can be used in the context of self-labeling for base methods that produce tree structures, rules or decision lists. It involves the number of rules in the decision lists (or equivalently the number of leaves in a decision tree) and expresses the *relative growth* in structure as:

$$\Gamma = |E^g|/|E^w| \tag{9}$$

where E^g is the set of rules produced by the self-labeling method (here the grey-box) and E^w is the set of rules produced by the baseline white box when using only labeled data. For this measure, a number much greater than one indicates that a major growth in the structure of the self-labeling method is needed when using the extra unlabeled data. In that case, the balance between interpretability and performance must be taken into account for further evaluation.

The second measure is more general and applicable to any model whose structure is formed by quantifiable explanation units (e.g. rules, prototypes, features, derived features, etc.). Here, this measure estimates the *simplicity* of the model according with the size of the structure in terms of number of rules. Although it is often presumed that the smaller the rule set the better, this is not necessarily a linear relation. The desired simplicity in terms of number of rules has a smooth behavior which can drop quickly. Therefore, we propose to measure simplicity through a generalized sigmoid function, since it allows to represent this relation with enough flexibility:

$$\Upsilon(|E^g|) = \phi(|E^g|, \theta_1, \theta_2, \lambda, \eta, \nu) \tag{10}$$

$$\phi(|E^g|, \theta_1, \theta_2, \lambda, \eta, \nu) = \theta_1 + \frac{\theta_2 - \theta_1}{(1 + e^{-\lambda(|E^g| - \eta)})^{1/\nu}} \tag{11}$$

where $\theta_1 = 1$ and $\theta_2 = 0$ represent the upper and lower asymptotes of the function respectively, λ is the slope of the curve, η regulates the shift over the x-axis and ν affects near which asymptote maximum growth occurs. In this way, a value of 1 indicates high simplicity and it decreases smoothly toward 0. A bigger λ would make the function less smooth and the value of η moves where the middle value of the function is obtained. A value of $\nu = 1$ makes no change in the curve, while $\nu < 1$ moves the growth toward the upper asymptote and $\nu > 1$ toward the lower one. Observe that both η and ν influence where 0.5 simplicity is obtained. In real application scenarios these parameters should be decided based on the criteria of domain experts. Given the diversity of our benchmark, we set $\lambda = 0.1, \eta = 30, \nu = 0.5$ for illustrating a general setting (see Fig. 2).

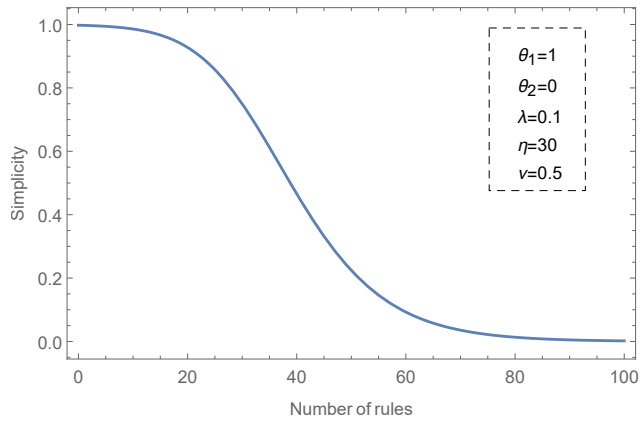


Fig. 2: Simplicity function with default parameters used for the benchmark datasets. For specific applications these parameters are domain dependent.

With these values, the function produces medium evaluations (around 0.5) when the number of rules is around 40. Similarly, it obtains rather high simplicity (higher than 0.8) when the number of rules goes below 30. However, parameter values should be estimated based on expert knowledge for specific applications. This highly flexible function allows customizing the value of simplicity according with the specifics of a given case study.

It is important to remark that the simplicity measure solely expresses what it would be considered a manageable model. Of course, a very simple model with only one rule and poor prediction rates is not desirable, whereas for a very simple dataset it might happen that three or four rules are enough to reach accurate results. That is why taking into account the prediction performance is fundamental for a proper assessment. In order to measure algorithms' quality based on the balance between the prediction rates and the simplicity of the learned model, we propose a third measure — called *utility* — combining the kappa and the simplicity values with a mixing parameter α,

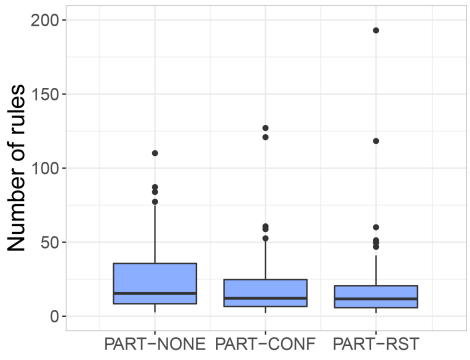
$$\Psi(E^g) = \alpha * \kappa(E^g) + (1 - \alpha) * \Upsilon(|E^g|) \qquad (12)$$

where α is set to 0.6 for our experimental setting, although other values are also possible.

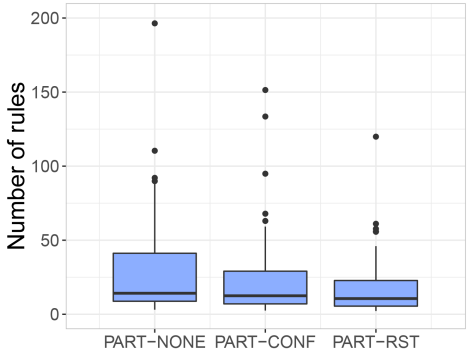
B. Evaluation of the Amending

In this section, we study how different choices of the amending processes impact the performance of SlGb. Therefore, we first explore the influence on the prediction rates. Table I shows slight improvements in the performance across each ratio. However, when examining the number of rules obtained, the difference is significantly visible. Figure 3 plots the number of rules produced by each combination, per ratio of labeled data. An interesting pattern is observed across ratios: RST-based amending further reduces the number of rules.

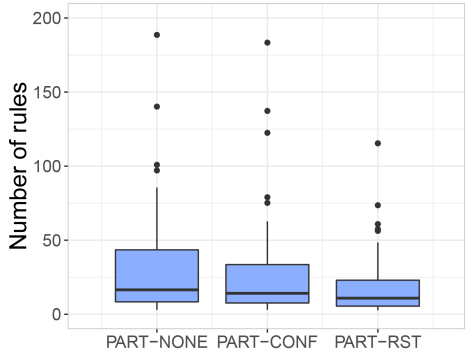
Table II shows the average relative growth and simplicity over the 55 datasets tested for the four ratios. Regarding the



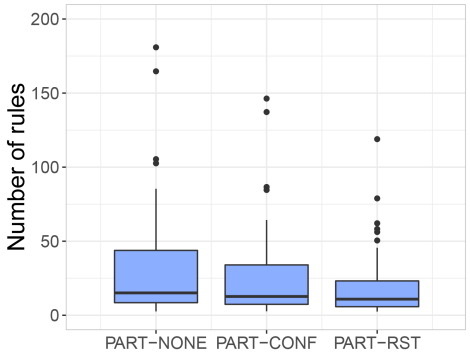
(a) Using 10% of labeled instances.



(b) Using 20% of labeled instances.



(c) Using 30% of labeled instances.



(d) Using 40% of labeled instances.

Fig. 3: Number of rules produced by each configuration. RST-based amending further reduces the number of rules.

TABLE I: Prediction rates by labeled ratio in terms of kappa mean (and standard deviation) achieved by SlGb with the proposed RST-based amending (RF-PART-RST) against SlGb using confidence based amending (RF-PART-CONF) and SlGb without amending (RF-PART-NONE).

	Ratio	10%	20%	30%	40%
RF-PART-NONE	mean	**0.56**	0.60	0.61	**0.62**
	(stdev)	**(0.28)**	(0.27)	(0.27)	**(0.27)**
RF-PART-CONF	mean	0.56	0.60	0.61	**0.62**
	(stdev)	(0.29)	(0.27)	(0.27)	**(0.27)**
RF-PART-RST	mean	**0.56**	**0.61**	**0.62**	**0.62**
	(stdev)	**(0.28)**	**(0.27)**	**(0.27)**	**(0.27)**

relative growth, the increase in the structure of the grey-box is on average larger when using small amounts of labeled data, while for bigger ratios this difference decreases. This growth in the structure is an expected consequence of providing more unlabeled data to the white-box surrogate in the grey-box scheme. However, the use of amending procedures alleviates this effect by giving more importance to relevant unlabeled instances. In general, a smaller growth is observed when using RST amending thus resulting in the winner combination for all ratios. In addition, in Table III the simplicity measure (the closer the value to one the better) also indicates that in general the use of amending is convenient for obtaining more concise sets of rules. For this measure the proposed RST-based amending exhibits the highest values of simplicity for all ratio values used for experimentation.

TABLE II: Interpretability by labeled ratio in terms of relative growth mean (and standard deviation) achieved by SlGb with the proposed RST-based amending against SlGb using confidence based amending and SlGb without amending.

	Ratio	10%	20%	30%	40%
RF-PART-NONE	mean	3.07	2.19	1.78	1.55
	(stdev)	(0.92)	(0.62)	(0.51)	(0.47)
RF-PART-CONF	mean	2.11	1.66	1.45	1.30
	(stdev)	(0.59)	(0.47)	(0.43)	(0.40)
RF-PART-RST	mean	**1.99**	**1.38**	**1.13**	**0.98**
	(stdev)	**(0.49)**	**(0.31)**	**(0.24)**	**(0.21)**

TABLE III: Interpretability by labeled ratio in terms of simplicity (and standard deviation) achieved by SlGb with the proposed RST-based amending against SlGb using confidence based amending and SlGb without amending.

	Ratio	10%	20%	30%	40%
RF-PART-NONE	mean	0.70	0.70	0.69	0.69
	(stdev)	(0.39)	(0.39)	(0.40)	(0.40)
RF-PART-CONF	mean	0.81	0.78	0.75	0.74
	(stdev)	(0.32)	(0.34)	(0.35)	(0.36)
RF-PART-RST	mean	**0.82**	**0.81**	**0.81**	**0.81**
	(stdev)	**(0.32)**	**(0.33)**	**(0.33)**	**(0.34)**

Fig. 4 visualizes the utility values in a heat-map plot. From this figure, it is easy to perceive that RST amending, positively contributes to the overall performance of the approach when taking both kappa and simplicity into account.

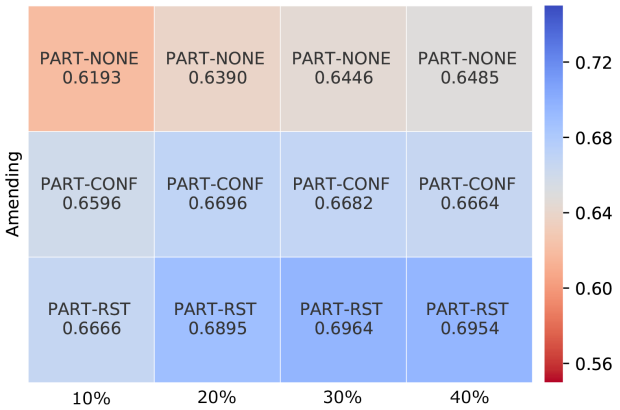


Fig. 4: Mean utility values for amending types across ratios. RST-based amending shows the best results for all ratios.

From the experiments above, we can conclude that the proposed RST amending contributes to obtaining more concise rules sets in the white box, without sacrificing performance of the grey-box model. This gain in simplicity is relevant as a proxy for interpretability in the form of transparency and simulatability of the final model.

C. Comparing against Self-labeling Classifiers

In this subsection, we compare the predictive capability of SlGb against the four best self-labeling techniques reported in the review paper in [16]: Co-training using support vector machine [1] (CT(SMO)), Tri-training using C45 decision tree [19] (TT(C45)), Co-Bagging using C45 decision tree [1] (CB(C45)) and Democratic Co-learning [20] (DCT). Since these algorithms are not inherently interpretable we focus our comparison on the prediction rates only. For this experiment, SlGb refers to the RF-PART-RST combination.

TABLE IV: Mean and standard deviation of kappa coefficient obtained by SlGb and four self-labeling methods from the state-of-the-art, by ratio. The best performance is highlighted.

	Ratio	10%	20%	30%	40%
SlGb	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)

Table IV reports the mean and standard deviation of kappa coefficient, reveling that our proposal has the highest mean for all ratios. There is no doubt about the superiority of the SlGb classifier when tested with datasets with ratios of 10% and 20% of labeled instances, as the differences are clearly visible. In the case of datasets comprising 30% and 40% of labeled

instances, the results show that SlGb is the best-performing classifier, but with less pronounced differences against DCT (for 30%) and CT(SMO) (for both ratios). However, DCT and CT(SMO) cannot be considered transparent due to their complex structure involving support vector machines and collaboration between base classifiers.

A more detailed analysis including statistical tests supporting these conclusions can be found in a extended study provided by authors as supplementary material [26]. Although our main goal was not to outperform the SSC methods in terms of classification rates, the analysis reported above supports our claim that we obtain a favorable balance between performance and interpretability by using the self-labeling grey-box approach for solving SSC problems.

V. Conclusions

In this paper, we have introduced a RST-based amending procedure for weighting the instances coming from the self-labeling process in the SlGb semi-supervised classifier. In addition, we proposed measures for rule-based classifiers in order to evaluate the simplicity of the final model as a proxy for interpretability. Numerical experiments using accuracy and interpretability measures show that RST-based amending produces more concise sets of rules without affecting the prediction rates by giving more importance to confident instances. In addition, the experimental comparison shows that our SlGb 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.

References

[1] M. F. A. Hady and F. Schwenker, "Co-training by committee: a new semi-supervised learning framework," in *Proceedings of the 2008 IEEE International Conference on Data Mining Workshops*. IEEE, 2008, pp. 563–572.

[2] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 1995, pp. 189–196.

[3] D.-H. Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on challenges in representation learning, ICML*, vol. 3, 2013, p. 2.

[4] M. Li and Z.-H. Zhou, "Setred: Self-training with editing," in *Proceedings of the 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, vol. LNCS 3518. Springer, 2005, pp. 611–621.

[5] J. M. Alonso, L. Magdalena, and G. González-Rodríguez, "Looking for a good fuzzy system interpretability index: An experimental approach," *International Journal of Approximate Reasoning*, vol. 51, no. 1, pp. 115 – 134, 2009.

[6] F. Doshi-Velez and B. Kim, "Considerations for evaluation and generalization in interpretable machine learning," in *Explainable and Interpretable Models in Computer Vision and Machine Learning*, H. J. Escalante, S. Escalera, I. Guyon, X. Baró, Y. Güçlütürk, U. Güçlü, and M. van Gerven, Eds. Springer, 2018, pp. 3–17.

[7] Z. C. Lipton, "The mythos of model interpretability," in *Proceedings of the 33rd International Conference on Machine Learning. Workshop on Human Interpretability in Machine Learning*, 2016, pp. 96–100.

[8] C. Molnar, *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. Christoph Molnar, 2019.

[9] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you?: Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 1135–1144.

[10] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 4765–4774.

[11] I. Grau, D. Sengupta, M. Garcia Lorenzo, and A. Nowe, "Interpretable self-labeling semi-supervised classifier," in *Proceedings of the 2nd Workshop on Explainable Artificial Intelligence, International Joint Conference on Artificial Intelligence IJCAI/ECAI 2018*, D. Aha, T. Darrell, P. Doherty, and Daniel Magazzeni, Eds., 7 2018, pp. 52–57.

[12] ——, "Grey-box model: An ensemble approach for addressing semi-supervised classification problems," in *Belgian-Dutch Conference on Machine Learning BENELEARN 2016*, 9 2016.

[13] K. P. Bennett and A. Demiriz, "Semi-supervised support vector machines," in *Advances in Neural Information Processing Systems 11*. MIT Press, 1999, pp. 368–374.

[14] A. Blum and S. Chawla, "Learning from labeled and unlabeled data using graph mincuts," in *Proceedings of the Eighteenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 19–26.

[15] A. Fujino, N. Ueda, and K. Saito, "Semisupervised learning for a hybrid generative/discriminative classifier based on the maximum entropy principle," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 3, pp. 424–437, 2008.

[16] I. Triguero, S. García, and F. Herrera, "Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study," *Knowledge and Information Systems*, vol. 42, no. 2, pp. 245–284, Feb 2015.

[17] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems 28*. Curran Associates, Inc., 2016, pp. 2234–2242.

[18] A. Halder, S. Ghosh, and A. Ghosh, "Ant based semi-supervised classification," in *Proceedings of the 7th International Conference on Swarm Intelligence*, M. Dorigo, M. Birattari, G. A. Di Caro, R. Doursat, A. P. Engelbrecht, D. Floreano, L. M. Gambardella, R. Groß, E. Şahin, H. Sayama, and T. Stützle, Eds., vol. LNCS 6234. Springer, 2010, pp. 376–383.

[19] Z.-H. Zhou and M. Li, "Tri-training: Exploiting unlabeled data using three classifiers," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 11, pp. 1529–1541, 2005.

[20] Y. Zhou and S. Goldman, "Democratic co-learning," in *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence*. IEEE, 2004, pp. 594–602.

[21] J. van Engelen and H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, 2019.

[22] Z. Pawlak, "Rough sets," *International Journal of Computer & Information Sciences*, vol. 11, no. 5, pp. 341–356, 1982.

[23] D. R. Wilson and T. R. Martinez, "Improved heterogeneous distance functions," *Journal of Artificial Intelligence Research*, vol. 6, pp. 1–34, 1997.

[24] R. Bello and J. L. Verdegay, "Rough sets in the Soft Computing environment," *Information Sciences*, vol. 212, pp. 1–14, 2012.

[25] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework." *Journal of Multiple-Valued Logic & Soft Computing*, vol. 17, 2011.

[26] I. Grau, D. Sengupta, M. M. G. Lorenzo, and A. Nowe, "An interpretable semi-supervised classifier using two different strategies for amended self-labeling," 2020, arXiv preprint arXiv:2001.09502.

[27] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[28] E. Frank and I. H. Witten, "Generating accurate rule sets without global optimization," in *Proceedings of the Fifteenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 144–151.

[29] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960.

[30] J. M. Alonso, C. Castiello, and C. Mencar, "Interpretability of fuzzy systems: Current research trends and prospects," in *Springer Handbook of Computational Intelligence*, J. Kacprzyk and W. Pedrycz, Eds. Springer Berlin Heidelberg, 2015, pp. 219–237.