Human-In-The-Loop Construction of Decision Tree Classifiers with Parallel Coordinates

Vladimir Estivill-Castro
Institute for Integrated and Intelligent Systems
Griffith University
Brisbane, Australia
v.estivill-castro@griffith.edu.au

Eugene Gilmore
Institute for Integrated and Intelligent Systems
Griffith University
Brisbane, Australia
eugene.gilmore@griffithuni.edu.au

René Hexel
Institute for Integrated and Intelligent Systems
Griffith University
Brisbane, Australia
r.hexel@griffith.edu.au

***Abstract*—How can there be Human-In-the-Loop-Learning (*HILL*) if datasets aimed at building classifiers have ever more dimensions? We make two contributions. First, we examine the few early results on the effectiveness of *HILL* for building autonomous classifiers and report on our own experiment that validates the merits of *HILL*. Second, we introduce a *HILL* system (by using parallel coordinates) for learning of decision tree classifiers (DTCs). DTCs importantly emphasise the relevance of attributes and enable attribute selection, and therefore are appreciated for their transparency. The proposed system addresses a number of the shortcomings of the many *HILL* systems and allows for easy exploration of datasets. In particular, we incorporate parallel coordinates effectively in our tool for visualisation of high dimensional datasets. We can not only focus the learning on the accuracy of classifiers, but we can enhance performance in other important factors such as system's interpretability and the ability to gain insight into datasets. Finally, we show the advantages of our *HILL* system in the application area of mobile robotics using the case study of image segmentation in robotic soccer.**

I. Introduction

The prevalence of Machine Learning (ML) in all application areas imaginable is increasing dramatically year after year. ML models usage is increasing because they can very quickly and accurately make predictions. While there has undoubtedly been considerable success in adoption of these methods, this does not come without its own problems. Many are starting to recognise that the high levels or accuracy and predictive power are coming at the cost of transparency and interpretability of the predictions of these models. Therefore, predictive accuracy is no longer the sole focus by data analysts and data scientists. Other criteria have emerged for evaluation of supervised learning methods [1], [2]. In classification tasks, the models often considered the best performers (because they can achieve such high levels of accuracy) are methods such as Convolutional Neural Networks (CNNs), ensembles and Support Vector Machines (SVMs). These types of models are considered to be "black box" models that can hardly be interpreted by the user [1], [3]. However, improving the comprehensibility of classification models is considered an important research direction [1], [2].

Some researchers have attempted to address this issues. One common strategy has been to first produce accurate black-box models and then try to find ways of explaining them [4], [5]. This strategy generally takes one of two forms. One approach is to produce surrogate interpretable models that try to closely match the output of the black-box model while focussing on measurable quantities such as cluster size for explanation [6] The second approach is to generate an explanation for a particular instance being classified [7] or instances belonging to a subset of the feature space [8].

Others have strongly argued that if we really want interpretability, we need to learn interpretable models to begin with [9]. We take this latter approach with our proposal of Human-In-the-Loop-Learning (*HILL*) of decision-trees. Unlike black-box models, one remarkable feature of decision trees is they are interpretable by human users [1], [10].

Moreover, under the name of Interactive Machine Learning (IML) [11], the knowledge and expertise of human professionals is elicited and hopefully transported using *HILL* for autonomous classifiers. That is, not only datasets are the source of knowledge but IML also taps into the experience of humans in the field [12].

Decision-tree learning is perhaps one of the earliest and most well-established methods of constructing a predictive model from supervised instances in order to map new instances (whose class is unknown) to a target value. This technology has become ubiquitous in statistics, data mining, and ML. In fact, a particular form under the name C4.5 [13] was listed first among the top 10 most used algorithms in data mining [14], but it is acknowledged that the underlying ideas originated much earlier with CLS [15] and ID3 [16]. Also among the top 10 algorithms in data mining appears CART (Classification and Regression Trees) [17], which is also a decision-tree approach.

In this work we aim to build a system to facilitate learning interpretable models with explainable outputs that are also accurate. To achieve this, we use decision trees in conjunction with two often overlooked techniques. One technique is the understanding gained by visualisation [18]. For visualisation we make use of parallel coordinates [19] which is highly effective at visualising high dimensional datasets. We believe that parallel coordinates have great, but relatively unexploited potential in the machine learning and explainable AI communities. The second technique we argue for is the potential insight that can be provided by human experts in the construction of the predictive model. There is so much to gain by incorporating *HILL* in machine learning tasks, even validation or new knowledge elicitation by contrasting with previous insights [20]–[22]. In this paper, we aim to steer research in these two directions. We first provide a more in-depth evaluation of the WEKA [22] package for *HILL* by re-assessing the experiment where humans build classifiers. We stress that such earlier evaluations did not use the much larger participant pool we report here. Second, we propose a new system using parallel coordinates for exploration of datasets and *HILL* construction of decision-tree classifiers (DTCs). This newly proposed system addresses a number of shortcomings in many other *HILL* systems. With this new system we emphasise the importance of not only accuracy but also the ability of a human user to contribute their domain knowledge to a model, understand a learnt classifier and gain insights into datasets that the classifier is learnt from.

In the next section, we review why there are applications where *HILL* can benefit computer-vision systems and introduce motivation for a case study in robotics. In Section III, we follow the motivation for said case study with a review of the most salient *HILL* systems. We find there is almost no experimental evaluation of the effectiveness of *HILL* for building autonomous classifiers. Therefore, in Section IV, we describe our own experiment, which has a much larger user pool than any earlier experiment of the same type. In Section V, we introduce a *HILL* system (by using parallel coordinates) for interactive learning of decision tree classifiers (DTCs). The proposed system address a number of the shortcomings of the *HILL* systems we review and discuss in Section III. Section VI explains the context of our case study and the results we obtain with our system.

II. Human vision and Computer Vision

Although there are many application areas that can benefit from *HILL*, we show the power of our approach in the field of mobile robotics. In mobile robotics, sensors that detect and recognise objects or landmarks are crucial for many robotic tasks. Versatile robots in human environments are required not only to identify their human partners, but also the objects that are associated with missions involving robots and human participants. For the recognition of all these objects, the advances in deep learning and convolutional neural networks (CNNs) have proven to be revolutionary [23].

However, there are several issues with applying deep learning techniques. Some of these issues respond to the possibility of robust attacks to computer vision systems based on deep learning [24]. Others are related to the need to reach explainability criteria. In any case, using deep learning and convolutional neural networks is usually computationally expensive on two fronts; firstly, the construction of the classifiers and vision segmentation networks out of large data sets and secondly, the actual execution of the deep learning classifiers may be costly on board robots. Techniques have thus been developed to scale down the deep learning classifiers.

We exemplify deep learning issues here through a case based on object tracking from the RoboCup soccer competition. Mackworth's [25] classical proposal that a soccer competition of teams of robots represents an unprecedented challenge to artificial intelligence architectures and computer vision highlights how far we are to this day in artificially matching human capabilities. In particular, the real-time computer vision challenge at RoboCup is elevated each year [26] with less colour coding and artificial landmarks in a progressively larger soccer field. Thus, competing teams have changed in tandem their vision pipelines and vision methods, from human engineered [27] to increased use of on machine learning. Deep learning approaches appeared as a validation phase after a colour segmentation phase [28]. Although more sophistication has been achieved in this practical setting, usage of CNNs still requires a separate object proposal method. More importantly, the quality of the object recognition system largely depends on the efficiency of the algorithm used to generate candidates for classification (whether CNNS are used for binary classification tasks [29] or to detect several relevant object categories [30], or to detect robots (humanoids) [29]). Obviously, ball recognition has received the most attention [31]. Ball-only CNNs had input size massively reduced to be ported to typical robots for the humanoid league [32]. Even the so-called Visual Mesh technique [33] that uses multiple scales to improve the performance of neural networks requires a method to propose sub-regions. The classifier performs a crucial role in proposing a sub-region. Therefore our case study focuses on *HILL* of classifiers to identify candidate locations of the desired object within the image.

We argue that human input is of key interest here, as RoboCup aims at matches involving humans and robots, and thus, the objects of relevance are colour-coded for human accessibility. For instance, white objects, such as line marks on the ground and goal posts are such that pixel colours have semantic meaning where humans can incorporate their knowledge that such objects are white.

III. RELATED WORK

Given decision trees' nature of easy interpretability, a number of different *HILL* systems have been proposed to learn DTCs. Ankerst et al. proposed `PCB` [21] and later introduced their bar visualisation *HILL* system [34]. We highlight a number of limitations of this approach. Firstly, when the dataset is represented by bars, the user can no longer see actual values of attributes or the magnitude of difference in values. We argue that the missing information on values and the lack of representation of relative difference between values could limit human users' ability to contribute human knowledge effectively. This is because such relevant aspects of the human domain knowledge are omitted. Secondly, the bar representation also limits any rules to strictly univariate splits. The user also can not see any potential relationship between attributes with this type of representation.

We argue here for the increased use of parallel coordinates [19] for *HILL* classification systems. Parallel coordinates were innovative in IML with an algorithm named Nested Cavities [35], [36] (or NC). The construction of classifiers with NC is similar to decision-trees, because both approaches follow *Conditional Focusing* [37, Figure 8.3] and recursive refinement [16, Page 152 Chaper 4] that results in a decision-tree structure [38, Page 407]. But, to the best of our knowledge, there are no user-focused evaluations of IML with NC.

In a parallel-coordinates visualisation, each attribute of the dataset is shown as a parallel axis. Each value of the dataset is then shown as a poly-line that crosses each axis at the normalised value for that attribute. Unlike most other visualisation techniques, parallel coordinates are not restricted to showing only a certain number of dimensions. As the number of dimensions grows, more coordinates are displayed by packing them on the side. Eventually, an extremely large number of dimensions (over 100) becomes unmanageable. However, as pointed out in the same source, one should be *sceptical* [39] about decisions being based on over 100 variables and expect them to be interpretable and understandable. This point illustrates our proposal: to use the computing power to suggest the attributes that shall be displayed (as parallel coordinates) and suggested to the human user whilst allowing the human to choose the window they are comfortable to inspect.

Although some in the machine learning community have investigated the use of parallel coordinates in this area, none seem to have realised its full potential. Teoh and Ma present a dataset visualisation and decision tree construction techniques called `StarClass` [40] and later `PaintingClass` [41]. `StarClass` represents a dataset using a star-coordinates system. Although star coordinates do allow for a user to visualise a number of attributes in a relatively small amount of screen real-estate, we argue that it suffers some of the same issues as bar visualisation. Since the position of each point in star coordinates is determined by the value of all attributes, users cannot easily determine a subset most relevant, influential or predictive, let alone the ranges of values in a subset of attributes that discriminate between classes. Thus, users are prone to miss separation boundaries between classes provided by few(or even one) attributes. However, these types of separations between classes are flagrantly visible with parallel coordinates.

Moreover, if users' domain knowledge is that they know a subset of influential attributes or they would like to explore a desirable subset, there is no possibility to represent this with star coordinates (except to project the data into the subset of attributes a priori, but then why bother with the other dimensions). Thus, it may not be surprising that `PaintingClass` [41] is an extension of `StarClass` [40] so the user can use a visualisation on parallel coordinates. However, despite the power offered by parallel coordinates for visualisation of high dimensional data, the authors restrict the use of parallel coordinates to only categorical attributes, while still using star coordinates for numerical attributes. `PaintingClass` uses a modified version of parallel coordinates for categorical attributes where categorical values are spread out a long the axis according to their order in the dataset. This produces a visualisation with unintended bias. The system also has no means of algorithmic support for the user. It could be argued that it is not *HILL*.

Choo *et al.* also argue for the use of parallel coordinates for classification using their `iVisClassifier` system [42]. This approach also relies on full human involvement in building classifiers. However, it does involve computer power in the construction by using linear discriminant analysis (LDA) to try to minimise the number of attributes. Therefore, the number of the parallel coordinates displayed is reduced to the most influential LDA features. However, by using LDA, the system essentially creates another type of human understandability barrier. It is difficult, if not impossible, to interpret the new LDA features. The system does offer a tool to mitigate this obstacle for its focus application of face-recognition. For each LDA feature, a heat-map visualisation over the image frame is produced. Sadly, these visualisation of LDA features do not seem to have any human interpretable semantics, and result in the `iVisClassifier` being too tailored for the task of front-human-portrait face-recognition.

Probably one of the most well known systems for IML and *HILL* of DTCs is the `UserClassifier` system, available as a package in WEKA [22]. The `UserClassifier` system, allows a user to construct a decision-tree classifier (DTC) by hand. The system visualises the dataset by showing two-dimensional scatter plots of up to two attributes at a time. To help the user decide what attributes to inspect, small bars for each attribute are included, showing the distribution of classes when sorted by that attribute. For each internal node in the tree a user makes a rule by selecting a region on the two-dimensional plot. The user can also see the current tree as a simple node link diagram in a different view.

IV. The Weka User Classifier

Despite increasing interest in the machine learning community to include human experts in the learning process of autonomous classifiers, the evaluation of IML or *HILL* systems appears to be severely limited. The only IML system that reports on an experiment measuring the effectiveness with human users is WEKA's `UserClassifier` system and the experiment involved only five participants [22].

To compare how effective humans are at learning decision-tree classifiers, we conducted a new experiment aiming to establish unequivocally the claims that *HILL* offers explainability and interpretability. Our focus is to test the most genuine system of the type, namely, the original WEKA `UserClassifier` system [22].

In our experiment, an invitation was sent out to university students to participate.There are several reasons why students are considered suitable participants for experimentation with not overly complex tasks [43].Among these reasons is that students are not individuals who have worked with one particular tool for a long time. Their inexperience provides a

TABLE I: Datasets used.

Dataset	Attributes	Classes	Instances (Train)	Instances (Test)
Iris (Example Dataset)	4	3	100	50
Letter	16	26	15000	5000
Satellite	36	6	4435	2000
Segmentation	19	7	210	2100
Shuttle	9	7	43500	14500
Waveform	40	3	500	4500

clean slate and a measure to asses the learning curve and difficulty of tools and instruments [43].However, our subjects did have some elementary background in machine learning and DTCs and were not completely novice. Participants in our experiment were undergraduate and masters students enrolled in a database systems course which included topics on data mining. Participants had completed an earlier course in intelligent systems which emphasised machine learning techniques.

We ran our experiment as a practical laboratory towards the end of the second course after they had recently completed the theory component of the course on data mining and DTCs. The participants were all given a document containing instructions and an installation of the WEKA software. None of the participants had used the WEKA software before. The instructions directed the students through the process of using the `UserClassifier` system tool for constructing a DTC using the Iris dataset. Students were encouraged to explore the software using this dataset before constructing any further classifiers. Once participants were familiar with the WEKA software and completed a strong classifier for the Iris dataset, they were asked to make a DTC for five additional datasets. Table I shows the details of these datasets.

The experiment was run over a number of sessions. After gaining feedback from users in some of the initial sessions, the letter dataset was omitted. Participants commented that the large number of classes made this dataset very tedious and time consuming to work with. Since participation was voluntary, subjects aborted their engagement because dealing with this dataset seemed much more laborious. Since we observed many incomplete classifications and frustrated participants, our analysis does not include the incomplete work with the letter dataset. We take this opportunity to highlight that this confirms that not all classification tasks are necessarily suitable for *HILL*. The letter dataset (which is an optical character recognition task) is a good example of this. There is perhaps only little insight that could be given by humans since eliciting human knowledge in this domain is more challenging than achieving the high-accuracy rates possible with artificial systems. Nevertheless, we stress that in many other areas transparency is crucial.

Table II displays a summary of the results. A total of 50 students took part in the experiment. Since not all participants completed all tasks for all of the datasets, we removed affected work from consideration. Work submitted by participants was excluded for the following reasons.

- Not completing all datasets

TABLE II: Results from five datasets.

Method	Metric (Average)	Dataset			
		Satellite	Segment	Shuttle	Waveform
Present user study	Accuracy %	74.39(34)	82.47(35)	95.06(32)	62.49(29)
	F1	0.68(34)	0.82(35)	0.51(32)	0.61(29)
	Tree Size	45.64(34)	23.91(35)	21.81(32)	25(29)
	Time(seconds)	1273(30)	599(32)	470(28)	492(26)
Original study results [22]	Accuracy %	80.72(5)	86.07(5)	99.89(5)	70.82(5)
	Tree Size	82.2(5)	31(5)	25.4(5)	27.4(5)
	Time(seconds)	3144(5)	1584(5)	1584(5)	1332(5)
J48 (default settings)	Accuracy %	85.2	91	99.95	71.87
	F1	0.83	0.91	0.84	0.72
	Tree Size	443	25	43	85

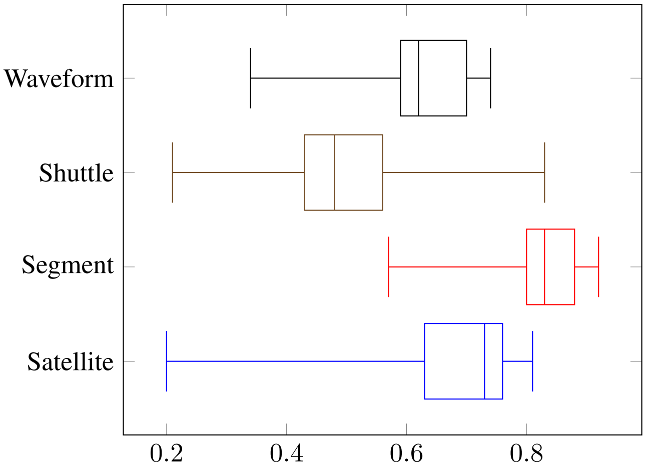


Fig. 1: Box-plots showing distribution of F1 scores for each dataset

- Not submitting the correct files
- Automatically learnt models (no human intervention)
- Using the class of the dataset as an attribute to test on
- Using the test set to train the model
- Using more than 2 hours to build a classifier.

Each item in the table also includes in brackets the number of data points for that result. Table II also includes results from the original study conducted by Ware et al. [22] Results from an automated DTC learning algorithm is also included as a baseline comparison. Along with overall accuracy, we include F1 scores for our results which take into account class imbalance issues.

Our results have some level of consistency with those reported in the original paper by Ware et al [22]. However, our results show that, in general, our participants grew slightly less accurate but smaller trees and in a shorter amount of time. We stress that the trade-off of shorter/smaller models, although less accurate, is a long debate in machine learning regarding the quality of the learning.

In any case, we believe these results establish the potential for *HILL* of DTCs. While the `UserClassifier` system is a good starting point, we argue that the interactions provided (on the data set and on the model) by the system limits the humans' ability to gain a broader understanding. We put forward the following limitations.

- The user can only see two attributes at a time, this is critically restrictive in the new world of big-data sets.
- The display of region bars is extremely small, making it difficult for users to decide what attributes to examine.
- There are no suggestions from the system of what attributes to select and where to create a split.
- Users can not visualise the tree unless they depart from the attribute visualisation window (losing context of the current splitting task).
- The visualised tree does not make use of colour, size, or any visualisation technique to communicate any properties of a node or an edge, or any relationship between a node and the dataset under analysis.

V. Constructing decision trees with parallel coordinates

We now present our system for *HILL* of DTCs. Our system aims to address the identified shortcomings of the *HILL* system examined in the previous section as well as the shortcomings of the systems examined in Section III. The proposed system uses parallel coordinates to visualise the training set. Figure 2 shows our system when visualising the Statlog (Landsat Satellite) [44] dataset.

Each attribute of the dataset becomes one axis in the parallel coordinates display. We represent each instance of the data by one of the coloured poly-lines in the visualisation, where the colour represents the class of the instance. On datasets with a large number of attributes, we enable the user to scroll left and right to shift a window of visible attributes. The system also gives the user the ability to change the order of axes as well as duplicate and flip axes. The system can suggest attributes to include in the visible window because of higher information gain or correlation with another attribute. All these operations are analogous to OLAP operations on a data-cube allowing users to visually explore and identify possible relationships between different attributes as well as the predictive power of attributes for classification. This last aspect is enabled by the concentration of colour.

Thus, our system offers visual exploration opportunities for the human's ability to spot patterns and to further investigate possible sophisticated relationships that are directly and transparently translatable to classification rules (see Section V-A). A user constructing a tree is simultaneously constructing a rule by selecting a range on one or more axes. As the user builds the classifier, the system displays the decision-tree on the left half of the screen. In the spirit of Hunt's generic recursive construction, the user can select any tree-node T to further refine (grow the tree deeper) from node T. Moreover, our system suggests nodes in two ways:

1) The colouring pattern of nodes provides feedback to the user about the purity of the node (and directly associates with accuracy of classification). The depth of node T in the tree is correlated with the generality and applicability of the rule derived from the path to T. Node depth also correlates to understandability and interpretability.

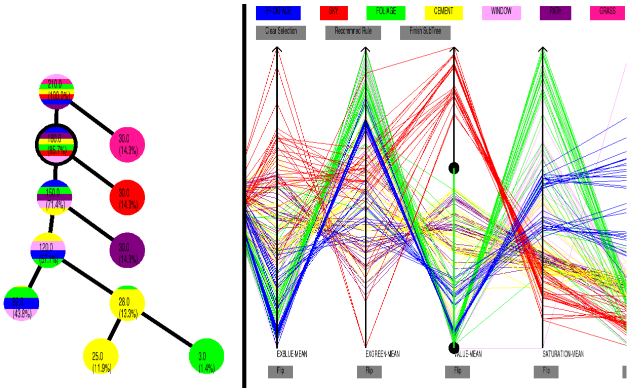


Fig. 2: Screenshot of the developed system being used to learn a decision-tree classifier for the satelite dataset

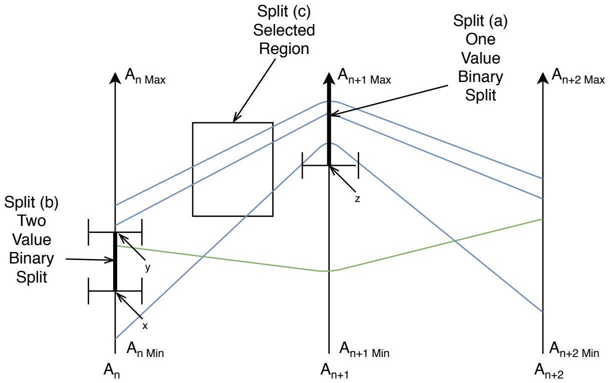


Fig. 3: Different splitting techniques. Split (a), a simple binary rule, selecting all instances that have a value greater than z for attribute A_{n+1}. Split (b), a range based rule, selecting all instances that have a value between x and y for attribute A_n. Split (c), our proposed region based split for a rule, selecting all instances that pass through the specified region between attribute A_n and A_{n+1}.

2) Values of predictability power of attributes, such as information gain, are delivered by the system.

A. Rules in the Plane

We also include in our system a type of rule-split proposed in our previous work [45]. This type of rule relies on the unique characteristics of the parallel coordinates plane for rule visualisation. Figure 3 shows the most common rule splitting techniques currently used as well as our proposed new method. Rather than selecting a split based on a particular axis, the split is described by a region in the parallel coordinates plot. The point-line duality in parallel coordinates dictates that a point in a parallel coordinates plot represents a line in the Cartesian plane [19]. If the split was based on a particular point in the parallel coordinates plane this shall be interpreted as a rule that requires points to follow closely some linear correlation between two attributes. Given that, in any real dataset, there is some distortion involved, the region-based construction of a rule is interpreted as a rule that requires points to follow closely some linear correlation between two attributes with also some margin for error. The margin of discrepancy from exact linear correlation has an interpretation as well, namely it is directly proportional to the the size of the region drawn by the user in the parallel-coordinates plot. We stress that all the drawn elements and their margins (in an analogous sense to support- vector-machine margins) in the visualisation have an interpretation as classification rules. This greatly contributes to the interpretability of the resulting classifier. Simultaneously, the involvement of only a few attributes (one or two attributes) contributes to the understandability of the rule (as opposed to full oblique test [46] which are linear combinations of all dimensions and hard to understand by humans [47]).

Therefore, our system offers several interesting effects when users manipulate the shape and size of the rectangle that defines the split and formation of a classification rule. We refer the reader to our previous work [45] for further discussion on these effects. Suffice to say that we stress here that our rule splits also allow for the increased expressivity over the axis parallel splits of traditional DTCs, because they offer oblique splits but only on a few and the most relevant attributes to the human user. This also has the advantage that the user can view the entire dataset or subset in the context of each rule.

B. *Algorithmic Support*

Similar to systems such as PCB [34], algorithmic support is included in our *HILL* system to assist the user with insights from metrics and indicators of machine learning algorithms. Whenever a user selects a node in the tree, only the subset of the dataset that matches the rules of all parent nodes is shown. Importantly we also order the parallel coordinates axes depending on the maximal information gain achievable using each attribute. By showing attributes first that the systems perceives as having the best potential for a good split we can effectively use both the human and the machine for learning. Each time a user selects a node for a split, the user can request to visualise our system's suggestions for a possible split on an attribute. These are splits from information gain, gini-index and other heuristics and strategies for fully autonomous construction of DTCs. The user can also request a suggestion of a parallel-coordinates region to define the rule, and the system calculates this using differential evolution [45]. Users can choose to use such machine-learning suggestions, modify a suggestion slightly, or disregard the suggestion completely and build their own rule. The user can also request that the system atomically grow a sub-tree from a selected node. The user can choose whether the rules produced in this generated sub-tree have the standard axis-parallel splits or take the form of our proposed parallel-coordinates region rules. This allows the user to interpret and validate the sub-tree and investigate the choices of the system for the attributes involved in the split. Thus, the user gains insights into the structure of the phenomena coded in the training set instances. Recall that one virtue of decision trees is that attributes higher in the tree have higher predictive power, while lower in the tree, they have conditional predictive power.

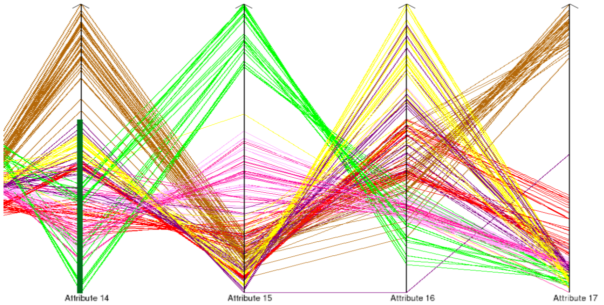


Fig. 4: An example of a situation where *HILL* can produce better models. The selected region on attribute 14 is the split recommended by the system while visually a user can quickly see a better alternative using attribute 17.

The user can then opt whether or not to use this generated sub-tree, but also we offer the option to use part of the generated sub-tree. The addition of this algorithmic support mechanism strikes a good balance between user insight and machine number crunching into the learning process. Without such a mechanism, manually learning the entire classifier would be tedious and time consuming. Nevertheless, the user always retains full control on how the tree is learnt. We argue that humans' advanced pattern-recognition capabilities and the user's ultimate say result in both high accuracy classifiers and more understandable classification.

We believe that in many cases, this human assistance can allow more accurate and intuitive models to be constructed. Figure 4, shows one example where it can be seen that the system recommends a rule to split out the brown class. Indeed, this is a split that results in the highest possible gain ratio. However, for a human, it is obvious that a better split is possible, since the visualisation highlights such a split in a subset of the data. As indicated in the figure, a different attribute can also separate out the brown class. Although a split on this attribute has the same gain ratio, since it is not a better gain ratio than the first split found, the system recommends the first split. Visually, it can be seen that this second split would likely be much better as the brown class is glaringly separate from any other class.

C. *Visualising The Tree*

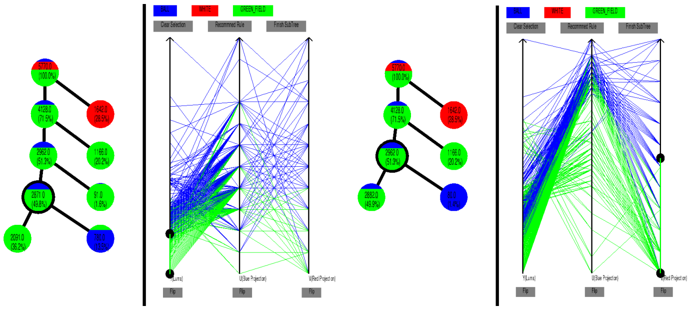
As the user constructs the DTC, they can always see the current state of the tree on the left half of the screen. This half of the screen is fully interactive and a user can click on any node in the tree. This will show the subset of training data reaching the node in the parallel-coordinates display, as well as any rules for this node. We also make use of colour and metrics to increase the user's understanding of the dataset and facilitate the model learning process. Each node in the tree shows a histogram reflecting the number of instances of each class in the training set that reached the node. Therefore, the size of a coloured area in a node is in direct proportion to the number of training instances of the corresponding class reaching that node. Thus, as we mentioned, users can quickly identify which nodes are sufficiently pure and do not need further refinement. We also show both the absolute number of training instances reaching this node as well as the percentage relative to the size of the training set.

VI. Robotics Case Study

We have argued (along with others [11], [12], [21], [22], [40], [41], [48]) that there are many application areas where *HILL* for machine learning is highly beneficial. We now present a case study from robotics to highlight some of the advantages to our approach using *HILL*. For our case study, we look at the open problem of image processing in real-time on board a NAO robot for the purpose of playing robotic soccer in the RoboCup SPL competition. Specifically, we look to address the problem of finding a soccer ball on the field using the NAO's cameras during a game. The current rules of the SPL specify that the ball is a standard pattern of black and white. A good way to identify candidate locations for a CNN to look for a soccer ball in an image is to detect the black sections of the ball. The ball is the only object on the field that has distinctive patterns of black. Thus, this is human knowledge to incorporate; namely, finding a sub-region of black pixels in an image suggests a high likelihood of a ball in that portion of the image. The difficultly comes in reliably detecting these black pixels. The SPL competition has progressed to allow for varying and more natural lighting conditions. This means that the dark green of the field can often provide pixels that appear black in an image. We propose using our *HILL* system to create a DTC to colour-segment pixels in an image with a particular focus on identifying black pixels of the ball.

We collected images using the NAO's cameras during a match of the SPL competition. Using specially designed software, humans labelled regions of each image indicating a dominant colour. Only three colour labels were used: green for pixels belonging to the field, black for the black spots of the ball, and white for the white spots of the ball and the field lines, with everything else remaining unlabelled. We configured a dataset for a vanilla machine learning classification task. This is, in fact, an image segmentation task where the raw pixels (their colour values) are the independent variables and the class label(green, black or white) is the dependent variable. The colours of each pixel were encoded as YUV values which makes up the three attributes of the dataset. The dataset contained 6,370 instances in total. The dataset was randomly shuffled and 600 instances were removed to be used as a test set. One of the advantages of our *HILL* system is that the human driver can specify what is most important in the model, whereas a traditional learning algorithm would only try to optimise for a quantitative goal such as overall accuracy.

We use our *HILL* system to create colour segmentation DTCs for two different situations. The first situation is where a soccer playing robot has no information about where the ball previously was and is not currently in any game-critical situation. In this case, the robot can afford to spend more time applying a CNN to more and larger sub-regions, hoping to find the ball. For this situation, we create a DTC that focuses on high recall of the black class i.e. the number of false negatives for the black class should be low. The second situation is where the soccer playing robot has previously seen the ball close by or is in a situation where it needs to prioritise CPU time for other tasks. In this situation, the main priority should be to reduce and limit the number of candidate sub-regions. We shall not waste resources on many sub-regions with little probability of holding the ball. For this second situation, we create a DTC that focuses on high precision of the black class i.e. the number of false positives for the black class should be low. The results demonstrate the efficacy of this approach. Figure 5a shows the first DTC constructed with the aim of high recall for the black class. This classifier manages a recall of 0.95 and precision of 0.85 for the black class. Figure 5b shows the second DTC constructed with the aim of high precision of the black class. This classifier manages a recall of 0.42 and precision of 1.0 for the black class.



(a) High recall DTC

(b) High precision DTC

Fig. 5: Screenshot of the developed system being used to learn a DTC that has either a high precision or high recall for the black class(coloured blue)

VII. Conclusion

We have argued for the benefits of bringing the human into the loop for learning models for classification tasks. With a *HILL* system not only can we take advantage of human users' advanced pattern recognition capabilities, we can also include their domain knowledge and expertise directly into the model. At the same time, we added transparency and understandability to the models that are created along with insight into the datasets they are trained from. We argue that decision trees, which are widely used within the machine learning community, are an outstanding model for *HILL* classification. They naturally lend themselves to easy human interpretation while maintaining the ability for competitive levels of accuracy. Despite the lack of attention that parallel coordinates have received by the machine learning community, we argue it is an effective method of visualisation for *HILL* tasks. We showed this potential by proposing a system for *HILL* of DTCs. Finally, the example of colour segmentation for a mobile robot highlights how bringing the human into the loop can greatly benefit the learning process.

Acknowledgment

We would like to thank the members of the MiPal team for their support.

References

[1] A. A. Freitas, "Comprehensible classification models: a position paper," *SIGKDD Explorations*, vol. 15, no. 1, pp. 1–10, 2013.

[2] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM computing surveys (CSUR)*, vol. 51, no. 5, pp. 1–42, 2018.

[3] A. Moore, V. Murdock, Y. Cai, and K. Jones, "Transparent tree ensembles," in *The 41st Int. ACM SIGIR Conf. on Research & Development in Information Retrieval*, ser. SIGIR '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1241–1244.

[4] Q. V. Liao, M. Singh, Y. Zhang, and R. K. Bellamy, "Introduction to explainable AI," in *Extended Abstracts of the 2020 CHI Conf. on Human Factors in Computing Systems*, ser. CHI EA '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–4.

[5] W. Samek and K.-R. Müller, "Towards explainable artificial intelligence," in *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Cham: Springer International Publishing, 2019, pp. 5–22.

[6] A. Blanco-Justicia and J. Domingo-Ferrer, "Machine learning explainability through comprehensible decision trees," in *Machine Learning and Knowledge Extraction*. Springer, 2019, pp. 15–26.

[7] W. Samek, A. Binder, G. Montavon, S. Lapuschkin, and K. Müller, "Evaluating the visualization of what a deep neural network has learned," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 28, no. 11, pp. 2660–2673, 2017.

[8] H. Lakkaraju, E. Kamar, R. Caruana, and J. Leskovec, "Faithful and customizable explanations of black box models," in *Proc. of the 2019 AAAI/ACM Conf. on AI, Ethics, and Society*, ser. AIES '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 131–138.

[9] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, pp. 206–215, 05 2019.

[10] J. Huysmans, K. Dejaeger, C. Mues, J. Vanthienen, and B. Baesens, "An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models," *Decision Support Systems*, vol. 51, no. 1, pp. 141–154, 2011.

[11] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza, "Power to the people: The role of humans in interactive machine learning," *AI Magazine*, vol. 35, no. 4, pp. 105–120, Dec. 2014. [Online]. Available: https://www.aaai.org/ojs/index.php/aimagazine/article/view/2513

[12] J. A. Fails and D. R. Olsen, "Interactive machine learning," in *Proc. of the 8th Int. Conf. on Intelligent User Interfaces*, ser. IUI '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 39–45.

[13] J. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann Publishers, 1993.

[14] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. McLachlan, A. Ng, B. Liu, P. Yu, Z.-H. Zhou, M. Steinbach, D. Hand, and D. Steinberg, "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol. 14, no. 1, pp. 1–37, January 2008.

[15] E. Hunt, J. Martin, and P. Stone, *Experiments in Induction*. New York: Academic Press, 1966.

[16] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Reading, MA: Addison-Wesley Publishing Co., 2006.

[17] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Monterrey, CA: Wadsworth and Brooks, 1984.

[18] C. Mues, J. Huysmans, J. Vanthienen, and B. Baesens, "Comprehensible credit-scoring knowledge visualization using decision tables and diagrams," in *Enterprise Information Systems VI*. Springer Netherlands, 2006, pp. 109–115.

[19] A. Inselberg, *Parallel Coordinates : Visual Multidimensional Geometry and its Applications*. NY: Springer, 2009.

[20] V. Estivill-Castro, "Collaborative knowledge acquisition with a genetic algorithm," in *9th Int. Conf. on Tools with Artificial Intelligence, ICTAI '97*. Newport Beach, CA, USA: IEEE Computer Society, November 3rd-8th 1997, pp. 270–277.

[21] M. Ankerst, C. Elsen, M. Ester, and H.-P. Kriegel, "Visual classification: An interactive approach to decision tree construction," in *Proc. of the Fifth ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, ser. KDD '99. New York, NY, USA: ACM, 1999, pp. 392–396.

[22] M. Ware, E. Frank, G. Holmes, H. M. A., and I. H. Witten, "Interactive machine learning: letting users build classifiers," *Int. J. Hum.-Comput. Stud.*, vol. 55, no. 3, pp. 281–292, 2001.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Computer Vision – ECCV 2016*. Cham: Springer International Publishing, 2016, pp. 21–37.

[24] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning visual classification," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2018, pp. 1625–1634.

[25] A. K. Mackworth, "On seeing robots," University of British Columbia, Vancouver, BC, Canada, Canada, Tech. Rep., 1993.

[26] K. Pulli, A. Baksheev, K. Kornyakov, and V. Eruhimov, "Real-time computer vision with OpenCV," *Commun. ACM*, vol. 55, no. 6, pp. 61–69, Jun. 2012.

[27] I. Schwarz, M. Hofmann, O. Urbann, and S. Tasse, "A robust and calibration- free vision system for humanoid soccer robots," in *RoboCup 2015: Robot World Cup XIX*. Springer International Publishing, 2015, pp. 239–250.

[28] D. Albani, A. Youssef, V. Suriani, D. Nardi, and D. D. Bloisi, "A deep learning approach for object recognition with NAO soccer robots," in *RoboCup 2016: Robot World Cup XX*. Cham: Springer International Publishing, 2017, pp. 392–403.

[29] M. Javadi, S. M. Azar, S. Azami, S. S. Ghidary, S. Sadeghnejad, and J. Baltes, "Humanoid robot detection using deep learning: A speed-accuracy tradeoff," in *RoboCup International Symposium*. Cham: Springer International Publishing, 2018, pp. 338–349.

[30] S. O'Keeffe and R. Villing, "A benchmark data set and evaluation of deep learning architectures for ball detection in the RoboCup SPL," in *RoboCup Int. Symposium*. Cham: Springer International Publishing, 2017, pp. 398–409.

[31] A. Gabel, T. Heuer, I. Schiering, and R. Gerndt, "Jetson, where is the ball? using neural networks for ball detection at robocup 2017," in *RoboCup 2018: Robot World Cup XXII*. Cham: Springer International Publishing, 2019, pp. 181–192.

[32] D. Speck, P. Barros, C. Weber, and S. Wermter, "Ball localization for robocup soccer using convolutional neural networks," in *RoboCup 2016: Robot World Cup XX*. Cham: Springer International Publishing, 2017, pp. 19–30.

[33] T. Houliston and S. K. Chalup, "Visual mesh: Real-time object detection using constant sample density," in *RoboCup 2018: Robot World Cup XXII [Montreal, QC, Canada, June 18-22, 2018]*, ser. Lecture Notes in Computer Science, vol. 11374. Springer, 2018, pp. 45–56.

[34] M. Ankerst, M. Ester, and H.-P. Kriegel, "Towards an effective cooperation of the user and the computer for classification," in *Proc. of the Sixth ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, ser. KDD '00. New York, NY, USA: ACM, 2000, pp. 179–188.

[35] A. Inselberg and T. Avidan, "Classification and visualization for high-dimensional data," in *Proc. of the sixth ACM SIGKDD Int. Conf. on Knowledge discovery and data mining*. Boston, MA, USA: ACM, August 20th-23rd 2000, pp. 370–374.

[36] P. L. Lai, Y. J. Liang, and A. Inselberg, "Geometric divide and conquer classification for high-dimensional data," in *DATA 2012 - Proc. of the Int. Conf. on Data Technologies and Applications*. SciTePress, July 25th-27th July 2012, pp. 79–82.

[37] E. Hunt, *Concept Learning — An Information Processing Problem*, second printing ed. New York: John Wiley, 1962.

[38] P. R. Cohen and E. A. Feigenbaum, *The Handbook of Artificial Intelligence, volume III*. Stanford, CA: HeurisTech Press, 1982.

[39] A. Inselberg, "Parallel coordinates: Visualization, exploration and classification of high-dimensional data," 2008, iII.14 Parallel Coordinates: Visualization, Exploration and Classification of High-Dimensional Data.

[40] S. T. Teoh and K. Ma, "Starclass: Interactive visual classification using star coordinates," in *Proc. of the Third SIAM Int. Conf. on Data Mining*, vol. 112. SIAM, 2003, pp. 178–185.

[41] S. T. Teoh and K.-L. Ma, "Paintingclass: Interactive construction, visualization and exploration of decision trees," in *Proc. of the Ninth ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, ser.

KDD '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 667–672.

[42] J. Choo, H. Lee, J. Kihm, and H. Park, "ivisclassifier: An interactive visual analytics system for classification based on supervised dimension reduction," in *2010 IEEE Symposium on Visual Analytics Science and Technology*, 2010, pp. 27–34.

[43] B. A. Mustafa, "An experimental comparison of use case models understanding by novice and high knowledge users," in *New Trends in Software Methodologies, Tools and Techniques - Proc. of the 9th SoMeT_10*, ser. Frontiers in Artificial Intelligence and Applications, vol. 217. IOS Press, September 29th - October 1st 2010, pp. 182–199.

[44] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[45] V. Estivill-Castro, E. Gilmore, and R. Hexel, "Constructing interpretable decision trees using parallel coordinates," in *Proceedings of the Artificial Intelligence and Soft Computing - 19th International Conference, ICAISC 2020 Part II*, ser. Lecture Notes in Computer Science, L. Rutkowski, R. Scherer, M. Korytkowski, W. Pedrycz, R. Tadeusiewicz, and J. M. Zurada, Eds., vol. 12416. Springer, October 12th-14th 2020, pp. 152–164, to appear in, Proc. of the 19th Int. Conf. on Artifical Intelligence and Soft Computing.

[46] S. K. Murthy, S. Kasif, and S. Salzberg, "A system for induction of oblique decision trees," *J. Artif. Int. Res.*, vol. 2, no. 1, pp. 1–32, Aug. 1994.

[47] E. Cantú-Paz and C. Kamath, "Inducing oblique decision trees with evolutionary algorithms," *IEEE Trans. Evolutionary Computation*, vol. 7, no. 1, pp. 54–68, 2003.

[48] N. T.D., T. Ho, and H. Shimodaira, "Interactive visualization in mining large decision trees," in *Knowledge Discovery and Data Mining, Current Issues and New Applications, 4th Pacific-Asia Conf. PADKK 2000*, ser. Lecture Notes in Computer Science, vol. 1805. Kyoto, Japan: Springer, April 18th-20th 2000, pp. 345–348.