Online Incremental Learning with Abstract Argumentation Frameworks*

H. Ayoobi[1,3,*], M. Cao[2], R. Verbrugge[1] and B. Verheij[1]

[1]Department of Artificial Intelligence, Bernoulli Institute, Faculty of Science and Engineering, University of Groningen, The Netherlands.

[2]Institute of Engineering and Technology (ENTEG), Faculty of Science and Engineering, University of Groningen, The Netherlands.

[3]Department of Computing, Faculty of Engineering, Imperial College London, United Kingdom

Abstract

The environment around general-purpose service robots has a dynamic nature. Accordingly, even the robot's programmer cannot predict all the possible external failures which the robot may confront. This research proposes an online incremental learning method that can be further used to autonomously handle external failures originating from a change in the environment. Existing research typically offers special-purpose solutions. Furthermore, the current incremental online learning algorithms can not generalize well with just a few observations. In contrast, our method extracts a set of hypotheses, which can then be used for finding the best recovery behavior at each failure state. The proposed argumentation-based online incremental learning approach uses an abstract and bipolar argumentation framework to extract the most relevant hypotheses and model the defeasibility relation between them. This leads to a novel online incremental learning approach that overcomes the addressed problems and can be used in different domains including robotic applications. We have compared our proposed approach with state-of-the-art online incremental learning approaches and an approximation-based reinforcement learning method. The experimental results show that our approach learns more quickly with a lower number of observations and also has higher final precision than the other methods.

Keywords

Argumentation-Based Learning, Online Incremental Learning, Argumentation Theory, General Purpose Service Robots

1. Introduction

This paper is a short version of our journal paper [1]. The development and application of domestic service robots are growing rapidly. Whereas basic household robots are already

ArgML2022: 1st International Workshop on Argumentation Machine Learning @COMMA 2022, September 13, 2022, Cardiff, Wales, UK

*This work is conducted at DSSC and sponsored by a Marie Skłodowska-Curie COFUND grant, agreement no. 754315.

*Corresponding author.

h.ayoobi@imperial.ac.uk (H. Ayoobi); m.cao@rug.nl (M. Cao); l.c.verbrugge@rug.nl (R. Verbrugge); bart.verheij@rug.nl (B. Verheij)

https://www.rug.nl/staff/h.ayoobi/research (H. Ayoobi); https://www.rug.nl/staff/m.cao/ (M. Cao); https://rinekeverbrugge.nl/ (R. Verbrugge); https://www.ai.rug.nl/~verheij/ (B. Verheij)

0000-0002-5418-6352 (H. Ayoobi); 0000-0001-5472-562X (M. Cao)

common practice [2], the study of General Purpose Domestic Service Robots (GPSR) able to do complex tasks is increasing [3, 4]. Due to the dynamic environment around GPSRs, they need to efficiently handle noise and uncertainty [5].

On the hardware level of GPSRs, any kind of system failure should be avoided. On a practical level, which involves persistent changes in the environment, it becomes much more difficult to account for all possible external failures at design time. Therefore, it is important to note that confronting unforeseen failures is mostly the default state for GPSRs, rather than an exceptional state as often described in the literature. There are some solutions for external failure recovery in the literature, which involve using simulations for the prediction of external failures [6] and logic-based reasoning to account for external failures [7, 8]. However, in most of these cases, the solutions are proposed for specific applications. In the following, we use the word "Failure" instead of the word "External Failure" for conciseness. This means that the focus of our research is not on system/hardware failures. In this paper, we propose an argumentation-based incremental online learning method for recovering from unforeseen failures.

1.1. Argumentation

Argumentation is a reasoning model based on interaction between arguments [9]. Argumentation has been used in various applications such as non-monotonic reasoning [10], inconsistency handling in knowledge bases [11], and decision making [12]. In [13], Dung has defined an Abstract Argumentation Framework (AF) as a pair of the arguments (whose inner structures are unknown) and a binary relation representing the attack relation among the arguments. Extending Dung's idea, some arguments can support a conclusion and others might be against (attacking) that conclusion in the bipolar argumentation framework [14]. Both the Bipolar Argumentation Framework (BAF) and the Abstract Argumentation Framework (AF) are used in the proposed argumentation-based learning approach.

1.2. Argumentation in Machine Learning

According to a recent survey by Cocarascu et al. [15], the works using argumentation in supervised learning are listed as follows. Argumentation-Based Machine Learning (ABML) [16] uses the CN2 classification approach [17]. This method uses experts' arguments to improve the classification results. The paper by Amgoud et al. [18] explicitly uses argumentation. There are other approaches for improving classification using argumentation in the literature [19].

Machine learning techniques have also been used for argumentation mining [20, 21, 22]. Bishop et al. combined argumentation with machine learning to prevent failure in deep neural network based break-the-glass access control systems [23].

In contrast with the aforementioned methods, we do not use argumentation for improving the current machine learning approaches or resolving conflicting decisions between current classification methods; instead, we focus on the development of an online incremental learning method. Moreover, the proposed method only uses class labels for the testing phase and not for the training. Therefore, it can be utilized in open-ended (class-incremental) scenarios as well [24].

2. Background

The Abstract Argumentation Framework (AF) and Bipolar Argumentation Framework (BAF) are the building blocks of the online incremental learning approach proposed in this paper. AF, BAF and online incremental machine learning algorithms are formally defined in this section.

2.1. Formal Definition of Abstract Argumentation Framework

An argumentation framework defined by Dung [13] is a pair $AF = (AR, R_{att})$ where AR is a set of arguments, and R_{att} is a binary relation on AR, i.e. $R_{att} \subseteq AR \times AR$. The meaning of $A\ R_{att}\ B$ is that A *attacks* B where A and B are two arguments. In order to define the *grounded extension* semantics in AF, which is used in the proposed learning method, some semantics should be defined first.
(**Conflict-Free**) Let $S \subseteq AR$. S is conflict-free iff there is no $B, C \in S$ such that B attacks C.
(**Acceptability**) An argument $A \in AR$ is *acceptable* with respect to a set S of arguments iff for each argument $B \in AR$: if B attacks A then B is attacked by at least one element of S.
(**Admissibility**) A conflict-free set of arguments S is *admissible* iff each argument in S is acceptable with respect to S.
(**Characteristic Function**) The *characteristic function* F_{AF} in an argumentation framework $AF = (AR, R_{att})$ is defined as follows:
$F_{AF} : 2^{AR} \rightarrow 2^{AR}$ and
$F_{AF}(S) = \{A|A \text{ is acceptable with respect to } S\}$.
(**Grounded Extension**) The *grounded extension* of an argumentation framework AF, denoted by GE_{AF}, is the least fixed point of F_{AF} with respect to set-inclusion [13]. Since F_{AF} is a monotonic function with respect to set inclusion [13], the existence of the fixed point for this function follows from the Knaster-Tarski theorem [25].

It can be proved that the grounded extension of the abstract argumentation framework utilized in the proposed argumentation-based learning method is the singleton admissible sets which do not have both incoming and outgoing edges.

2.2. Formal Definition of an Abstract Bipolar Argumentation Framework

An Abstract Bipolar Argumentation Framework (BAF) [14] is an extension of Abstract Argumentation Framework by adding a support relationship. A BAF is a triple of the form $< AR, R_{att}, R_{sup} >$ where AR is the finite set of arguments, $R_{att} \subseteq AR \times AR$ is the *attack* set and $R_{sup} \subseteq AR \times AR$ is the *support* set. Considering A_i and $A_j \in AR$, then $A_i\ R_{att}\ A_j$ means that A_i attacks A_j and $A_i\ R_{sup}\ A_j$ means that A_i supports the argument A_j.

The semantics of BAF are as follows:
(**Conflict-Free**) Let $S \subseteq AR$. S is conflict-free iff there is no $B, C \in S$ such that B attacks C.
(**Admissible set**) Let $S \subseteq AR$. S is admissible iff S is conflict-free, closed for R_{sup} (if $B \in S$ and $B\ R_{sup}\ C \Rightarrow C \in S$) and S defends all its elements.
(**Preferred extension**) The set $E \subseteq AR$ is a preferred extension iff E is inclusion-maximal among the admissible sets. An inclusion-maximal set among a collection of sets is a set that is not a subset of any other set in that collection.

(**Supporting Weights**) Like [26] the support relations in our model also have an assigned weight. Therefore, a node with higher sum of supporting weights can attack nodes with lower sum of supporting weights.

2.3. Formal Definition of Online Incremental Machine Learning Algorithms

We define an incremental learning approach that uses a sequence of data instances $d_1, d_2, ..., d_t$ for generating the corresponding models $M_1, M_2, ..., M_t$. In case of incremental online learning, each data instance d_i incrementally updates the model and $M_i : \mathbb{R}^n \rightarrow \{1, ..., C\}$, where C is the number of class labels, is representing the model which depends on M_{i-1}. The online learning is then defined as an incremental learning which is also able to continuously learn. Incremental learning approaches have the following properties:

- The model should adapt gradually, i.e. M_i is updated using M_{i-1}.
- The previously learned knowledge should be preserved.

A recent study on the comparison of the state-of-the-art methods for incremental online machine learning [27] shows that Incremental Support Vector Machines (*ISVM*) [28, 29] together with LASVM [30], which is an online approximate SVM solver, and Online Random Forest (*ORF*) [31] outperform the other methods. The comparison methods used in our paper have been chosen based on the aforementioned survey [27].

The proposed argumentation-based incremental learning approach uses the bipolar argumentation framework to model the visited data instances and generate relevant hypotheses. Subsequently, the abstract argumentation framework is used to model the defeasibility relations (i.e. the attack relations) between the current set of generated hypotheses and predict the best action (recovery behavior) for an unforeseen incoming data instance. Furthermore, the model incrementally gets updated as new data instances enter the model.

3. Scenarios

The performance of the different methods is tested using two test scenarios. The aim of the first test scenario is to model a situation where a programmer has provided an initial solution (e.g., a top level behavior such as entering the room), while (s)he has not accounted for all possible failures (e.g., objects and persons blocking the entrance), but allows the robot to find new solutions whenever a (previously unseen) failure occurs.

The basic setup of the first test scenario is illustrated in Fig. 1. The high-level behavior of the robot aims to proceed from the initial location to the target location using three entrances. Different obstacles might be on its way to the target location. In these scenarios, an agent observes all the obstacle locations at once and chooses a single recovery behavior (action) for recovering from that failure state. The agent can reach the goal if it chooses the best recovery behavior; otherwise, it fails to reach the goal.

3.1. Recovery Behaviors

Whenever the robot is confronted with a failure state, it may use any of the following recovery behaviors to resolve the issue. The run-time of each recovery behavior in seconds is presented

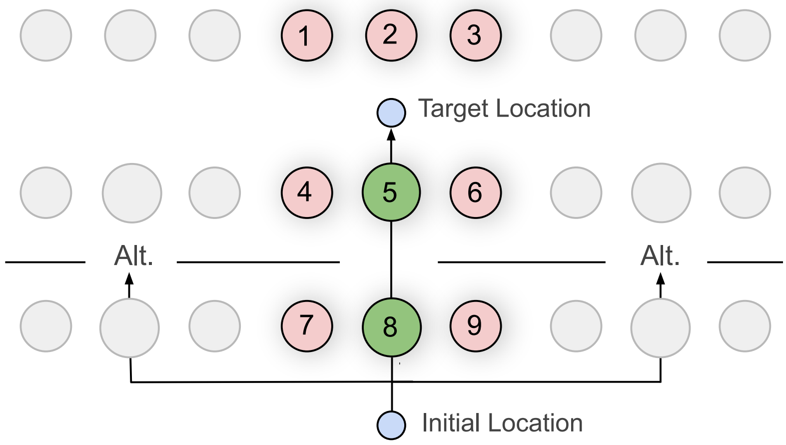


Figure 1: Schematic overview of the possible failure state scenario. Only the green locations are relevant for finding the best recovery behavior. Alt. stands for the Alternative Route recovery behavior.

in parentheses in front of each recovery behavior:

- Continue (2s): This solution is only useful if the failure has resolved itself (e.g., the obstacle moved away just after the failure).
- Push (5s): The robot can try pushing any obstacle.
- Ask (4s): The robot can try to ask any type to move.
- Alternative Route (Alt) (10s): The robot can move to another entrance to reach the target location.

It is important to note that choosing Alternative Route as the best recovery behavior may not always lead to success, because the robot may again be confronted with new obstacles (Fig. 1). Moreover, the best recovery behavior not only depends on the run-time of each recovery behavior, but also on the type, the color and the location of the obstacles.

3.2. Test Scenario 1

In this scenario, three types of obstacles (ball, box or person) with four colors (red, blue, green or yellow) can be presented in one of the locations 1 to 9 (Fig. 1). There can be either zero or one combination of color-type in each location. Only location number 5 and 8, marked in green (Fig. 1), is relevant for choosing the best recovery behavior. It is important to notice that the robot does not know this fact and it should infer that the only effective locations are location number 5 and 8 by observing different failure states in the environment. The agent observes all the obstacle locations at once and chooses a single recovery behavior (action) at each state. A new state is generated randomly at each time step. The number of possible combinations of the color-type in each location is 13 (3 types × 4 colors + "no obstacle" = 13). Since there are 9 locations in this scenario, the number of all possible states in this scenario is $13^9 = 10,604,499,373$.

3.3. Test Scenario 2

The second scenario has a different purpose and context. It shows the applicability of the proposed method outside the robotics field. The recent study on online incremental machine learning techniques [27] used the publicly available datasets from the UCI machine learning repository [32]. We also used the SPECT heart dataset from the UCI machine learning repository.

4. Method

In this section, we will discuss the proposed argumentation-based learning method for recovering from an unforeseen failure state[1].

4.1. Argumentation-Based Learning (*ABL*)

In order to explain *ABL*, we first use a simplified version of the previous test scenarios where there is only one location ahead of the robot (instead 9). When there is no obstacle ahead of the robot, the best recovery behavior is "Continue".

Assume that the robot confronts a blue-ball blocking the entrance. Since there is no pre-trained model yet, the robot tests different recovery behaviors in order of their run-time to find the best one. Supposing that pushing the ball was successful in this case, the robot should learn from this experience.

However, unlike the traditional tabular reinforcement learning techniques, only learning the best recovery behaviors (actions) for exactly the same experiences (states) is not enough. We need a learning approach capable of inferring the correlated feature values (each feature value is the color or type of the obstacle at each location or an empty location with no color and type) for choosing the best recovery behavior. This is known as *generalization* in the machine learning literature. For instance, confronting a red ball and a green ball with the same recovery behavior of pushing, the robot should make a new hypothesis *push a ball*. Therefore, the next time the robot confronts the yellow ball, it can easily infer that *Push* is the best recovery behavior.

Confronting a yellow ball with *Alternative Route* as the best recovery behavior contradicts the previous hypothesis. Therefore, a new hypothesis is made: *Push a ball unless it's yellow*. From an argumentation perspective, we can see each hypothesis as an argument. Therefore, the second generated hypothesis can attack and defeat the first argument. This is inspired by human agents who make new hypotheses from their perceptions and reason about the best course of action at each state.

The architecture of the proposed argumentation-based learning method is shown in Fig. 2. A bipolar argumentation framework is used as hypotheses generator unit and an abstract argumentation framework models the defeasibility relation between these generated hypotheses.

When a new data instance enters the model, all the combinations of its feature-values and the set of nodes in the grounded extension of the *AF* will be extracted. Each node (argument) in the AF unit is of the form *precondition → post-condition: weight*. According to the similarity between the *preconditions* of the arguments in the grounded extension and the feature values combinations, there will be three possible cases. Either there will be a unique similarity,

[1]See [1] for a more detailed and a more formal definition of the proposed method.

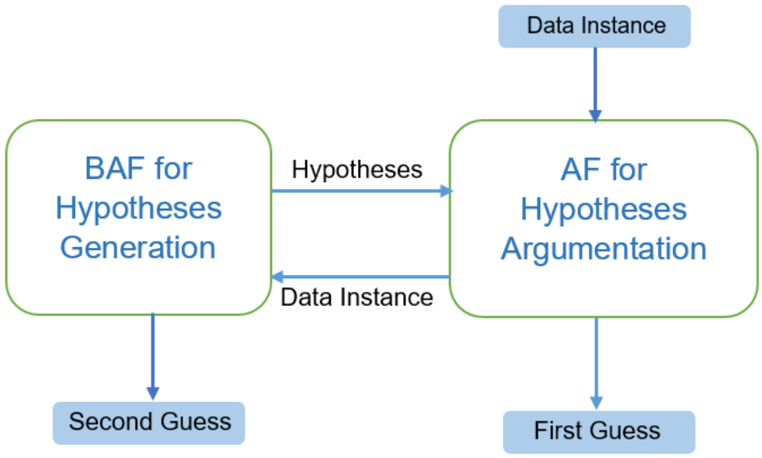


Figure 2: Architecture of the proposed Argumentation-based learning method.

multiple similarities or no similarity. In case of unique similarity, the post-condition of the argument (which is a recovery behavior) will be used as the first guess and will be applied to the environment to see the result. On the occasion that there exist multiple similarities, the recovery behavior with the highest weight among the arguments will be chosen and its post-condition will be applied to the environment. A successful recovery from the failure state will update the *BAF* unit. On the other hand, failure from recovery will lead to generating the second guess, updating the *BAF* unit, generating hypotheses from *BAF* unit and updating the *AF* unit, respectively.

We now use an illustrative example to explain the proposed method in more detail.

4.2. Example

Table 1 shows the best recovery behavior when the robot confronts an obstacle with different colors and types. Notice that this table is only used for this example and a randomly generated table is utilized for each of the 1000 independent runs for the experiments. Figure 3 shows the updating procedure of the model step by step. In the hypotheses generation unit (*BAF*), an arrow $\rightarrow$ shows a support relation between arguments and $\nrightarrow$ shows an attack relation between them. However, in *AF*, $\rightarrow$ shows an attack relationship between the arguments.

Referring to Table 1, at the beginning of the learning procedure, the robot confronts a Red-Ball (R-Ba). It tests all the recovery behaviors in order of their run-times and finds the *Push* recovery behavior as a success (Table 1). Subsequently, the Bipolar Argumentation Framework is getting updated as in Fig. 3. In order to update the *BAF*, first, the best recovery node is added which is *Push* in this case. Then all the possible combinations of the feature-values of the current state are added as supporting nodes. The supporting nodes for *Push* are *R*, *Ba* and *R-Ba*. If there previously exists the same supporting node, its supporting weight will be increased. For instance in Fig. 3, where *8:B-Bo* enters the *BAF*, since *B* and *B-Bo* are new supporting nodes for the *Alt (Alternative Route)* recovery behavior, they are added to the model with a supporting

Order	Color	Type	Best Recovery Behavior
1	Red	Ball	Push
2	Red	Box	Alternative Route
3	Red	Person	Ask
4	Green	Ball	Push
5	Green	Box	Alternative Route
6	Green	Person	Ask
7	Blue	Ball	Push
8	Blue	Box	Alternative Route
9	Blue	Person	Alternative Route
10	Yellow	Ball	Push
11	Yellow	Box	Alternative Route
12	Yellow	Person	Ask
13	None	None	Continue

Table 1
Possible combinations of color-type with the best recovery behaviors.

weight equal to 1. On the other hand, *Bo* already exists in the set of supporting nodes for *Alt* and its weight is increased. After updating the supporting weights, a set of hypotheses is generated based on the number of occurrences of each supporting node. For instance, after observing 1:Red-Ball (R-Ba), $R \rightarrow Push$ and $Ba \rightarrow Push$ are added to the AF unit.

Confronting *2:R-Bo* and using the previously generated hypotheses (specifically $R \rightarrow Push$), the robot would infer that the best possible recovery behavior is *Push*, which is a wrong choice in this case (Table 1). Therefore, the robot tries other recovery behaviors and finds *Alt* as success and updates the model accordingly. Moreover, a bidirectional attack will be added among all the recovery nodes in the *BAF* (in this case, *Alt* and *Push*). Subsequently, the new set of hypotheses is generated to update the hypotheses argumentation unit. Finally, an abstract argumentation framework is updated to model the attack relations between the set of generated hypotheses (arguments). This *BAF-AF* update cycle goes on and on during the learning procedure.

In this small example, seven out of thirteen predictions of the model are correct, and only two are wrongly classified using the proposed argumentation-based learning. In other cases, our system can provide multiple probable guesses. For instance, when *12:Y-P* enters the system in Fig. 3, the *AF* cannot provide any suggestion but the *BAF* will suggest both *Ask* and *Alt* as the candidate recovery behaviors. However, the mapping of the states to the best recovery behavior is randomly generated in all the experiments.

4.3. Hypotheses Generation Unit (BAF Unit)

This unit has two roles. Firstly, it generates a new set of hypotheses whenever the *AF* unit could not classify the new data instance correctly (1). The second role of this unit is to produce a second guess for the best recovery behavior (2):

1) In order to generate a new set of hypotheses from the constructed *BAF*, only one recovery behavior is considered which is highlighted with a red box in Fig. 3.

The only nodes which are getting updated during this process are the best recovery behavior

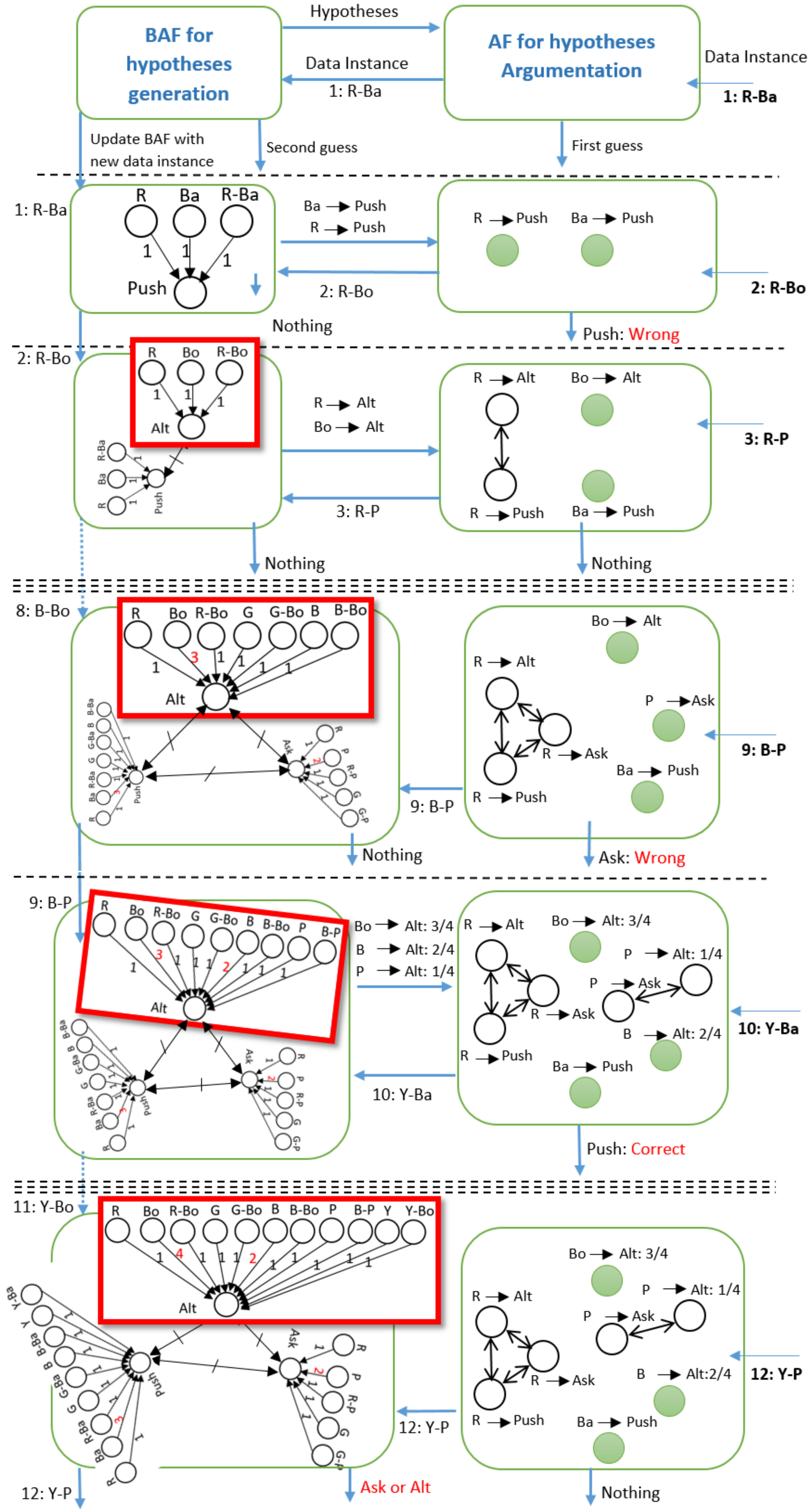


Figure 3: Example of Argumentation-Based Learning for the example scenario. Here only observations number 1, 2, 3, 9, 10 and 12 of Table 1 are shown selectively.

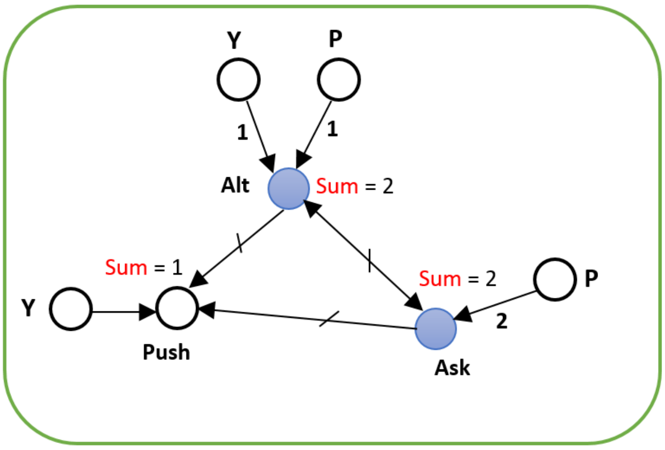


Figure 4: The generated *BAF* when Yellow-Person (*12:Y-P*) enters the model. Blue nodes show the intersection of preferred extensions and recovery behavior nodes.

for the current data instance and its supporting nodes. Autonomously identifying the best recovery behavior through trial and error, the update procedure for hypotheses generation takes place. The updating procedure searches for a node in the *BAF* graph with the best recovery behavior and appends all the possible combinations of the feature-values of the current state to the support nodes of the best recovery behavior node. In case that a supporting node already exists in the best recovery behavior node, its supporting weight is incremented.

2) In order to generate a second guess, a new *BAF* should be constructed. For an unforeseen failure state, the set of all possible combinations of feature-values is compared with the supporting nodes of each recovery behavior node. According to the sum of the matching supporting weights, the attack relations are adapted among the recovery behaviors. Therefore, only recovery behaviors with a higher sum of the matching supporting weights can attack the other recovery behavior. For instance, in the example, when *12: Y-P* enters the model for prediction, the *AF* is not be able to guess the best recovery behavior. Constructing a new *BAF* for a second guess, shown in Fig. 4, the calculated weighted sum for the *Alternative Route (Alt)* node is the same as *Ask* and higher than *Push*. Accordingly, the attack relations get updated. Using preferred extension semantics and its intersection with recovery behavior nodes, both *Alternative Route (Alt)* and *Ask* are chosen as the second guesses.

4.4. Hypotheses Argumentation Unit using *AF*

As stated in the previous sections, this unit tries to justify what has been learned so far by updating the attack relations between the arguments (hypotheses). The arguments in this framework can only bidirectionally attack each other when they have the same preconditions but different post-conditions.

When a new data instance enters the model, there are three possible cases for the set of hypotheses in the grounded extension of the *AF*. When the grounded extension of the *AF* is the empty set, the second guess is generated by the *BAF* unit. If one argument with the same post-condition exits in the grounded extension of the *AF*, then this post-condition will be the *AF*'s first guess. If more than one argument with different recovery behaviors in their post-condition was chosen, the weights of arguments determine which argument has more power to be selected. For instance in the example, if blue-ball enters the model after it has been trained using the complete set of data in Table-1, both $B \rightarrow$ *Alt: 2/4* and $Ba \rightarrow$ *Push:1* can be used for prediction. Since the $Ba \rightarrow$ *Push:1* has higher weight, the *Push* recovery behavior will be chosen, which is the correct choice for this failure state. Notice that in the proposed argumentation-based learning method, it can be proved that the grounded extension is a set of the singletons in the *AF*.

5. Experiments

In this section, we compare the performance of our proposed *ABL* method with other incremental learning techniques and an approximation-based reinforcement learning algorithm. The survey by V. Losing et al. compared a broad range of incremental online machine learning techniques [27]. Using the key methods in their survey, we are also comparing the proposed method with Incremental Support Vector Machine (*ISVM*) [28, 33, 34], incremental decision tree based on *C4.5* [35] and ID3, incremental Bayesian classifier [36], Online Random Forest (*ORF*)[31] and Multi-Layer Neural Networks for classification with localist models like Radial Basis Functions (*RBF*) which work reliably in incremental settings [37, 38].

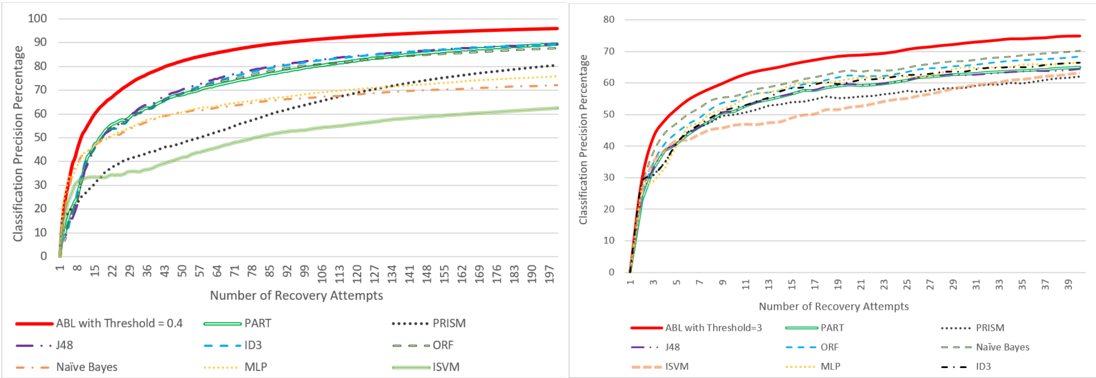
5.1. Comparison criteria

In the robotic scenario, we need a learning approach which can quickly learn to recover from failure states in a low number of attempts. Moreover, for the other test scenario, the goal is to incrementally learn from a lower number of training instances. Therefore, the increase in learning precision in a lower number of attempts is one important criterion (which we call *learning speed*) to evaluate the efficiency of the method [39]. Therefore, learning curves with the highest steepness in a smaller number of attempts are desirable. Furthermore, the *final learning precision* is also an important criterion.

5.2. Results

As one can see in Fig. 5a and Fig. 5b, the proposed Argumentation-Based Learning (*ABL*) method outperforms all the other methods in both the comparison criteria used for this research, namely, the final learning precision and the learning speed. The steepness of the learning curve shows that the *ABL* learns faster in a lower number of iterations.

For the first test scenario, after observing 30 failure states, *ABL* achieves 74% precision, while the best method among others has 60% precision. The final precision of *ABL* is 95%, while the best final precision among other methods is 90%.
In the second scenario, which differs from the prior scenario in context, *ABL* repeatedly outperforms all the other methods in both of the comparison criteria. Among other methods, incremental naive Bayes and incremental random forest (*ORF*) have better results. The final learning precision of *ABL* in this scenario is 75% while it is 70% for the incremental naive Bayes



(a) First Scenario.

(b) Second Scenario

Figure 5: Comparison of Argumentation-Based Learning (*ABL*) with key methods for incremental online learning [27] using a) the first test scenario b) the second test Scenario.

method. The slope of the learning curve also shows the faster learning speed of *ABL* with respect to all of the other methods.

6. Discussion

A key reason that the proposed method works better than Naive Bayes originates from the independence assumption between all features in the Naive Bayesian formulation. In the case of neural networks, considering that there is only a small number of training data instances, a complex neural network tends to over-fit and a small neural network leads to under-fitting. Choosing the best neural network architecture dynamically according to the number of visited data is also a challenging task. On the other hand, decision-tree based techniques fail at the initial recovery attempts and then gradually learn the best recovery behavior. This is because of the change in entropy or information gain when new unforeseen data updates the decision tree. This is also the case with the Online Random Forest (*ORF*) method. Furthermore, *ISVM* does not perform well in circumstances where only a few features are associated with predicting the class label. In all the above cases, the suggested *ABL* approach performed better as it considers any possible dependence between features and it can immediately focus on features which are most relevant for the optimal decision.

Moreover, *ABL* leads to an explicit representation of the learning process understandable for humans, as is also the case with decision-tree based techniques. In contrast, neural networks, support vector machines and Bayesian techniques are all black boxes [40] (this means that the trained models are not easily interpretable and explainable) for the humans. This explicit representation of the learning process can be utilized in combination with human-robot interaction. Employing this property, *ABL* can be used in multi-agent scenarios where agents can transfer their knowledge to each other.

Consequently, the proposed argumentation-based incremental learning algorithm could learn in fewer attempts with higher precision than other algorithms used for comparison. Moreover, ABL extracts an explicit set of rules that explain the knowledge acquired by the agent over the interaction with the environment. This feature makes the method more explainable and easy to debug by an expert.

Therefore, this method can be a good alternative when the feature values are discrete. Although we have shown that the current ABL approach is working well for the aforementioned scenarios in this paper, these results are limited to datasets with discrete feature values that are not high-dimensional. To make ABL more efficient for higher dimensional problems, we have introduced Accelerated Argumentation-Based Learning (AABL) [41] to improve the space and computational complexity of the method.

7. Conclusion

General purpose service robots should be able to recover from unexpected failure states caused by environmental changes. In this article, an argumentation-based learning (*ABL*) approach is proposed which is capable of generating relevant hypotheses for online incremental learning scenarios. This set of hypotheses is updated incrementally when unforeseen data enters the model. The conflicts among these hypotheses are modeled by Abstract Argumentation Frameworks.

The performance of *ABL* has been evaluated using both the robotics and the non-robotics incremental learning scenarios. The second scenario, which has a non-robotic context, is a publicly accessible dataset from the UCI machine learning repository. This scenario shows the fact that the proposed *ABL* method can be used in any online incremental learning application with discrete feature values. According to these experiments, the proposed method learns faster and with higher ultimate classification precision than various state-of-the-art online incremental learning methods.

References

[1] H. Ayoobi, M. Cao, R. Verbrugge, B. Verheij, Argumentation-based online incremental learning, IEEE Transactions on Automation Science and Engineering (2021) 1–15. doi:10.1109/TASE.2021.3120837.

[2] V. N. Lu, J. Wirtz, W. H. Kunz, S. Paluch, T. Gruber, A. Martins, P. G. Patterson, Service robots, customers and service employees: what can we learn from the academic literature and where are the gaps?, Journal of Service Theory and Practice (2020).

[3] M. Mende, M. L. Scott, J. van Doorn, D. Grewal, I. Shanks, Service robots rising: How humanoid robots influence service experiences and elicit compensatory consumer responses, Journal of Marketing Research 56 (2019) 535–556.

[4] S. Schneider, F. Hegger, A. Ahmad, I. Awaad, F. Amigoni, J. Berghofer, R. Bischoff, A. Bonarini, R. Dwiputra, G. Fontana, et al., The RoCKIn@Home Challenge, in: ISR/Robotik 2014; 41st International Symposium on Robotics, 2014, pp. 1–7.

[5] Y. Jiang, N. Walker, J. Hart, P. Stone, Open-world reasoning for service robots, in: Proceedings of the International Conference on Automated Planning and Scheduling, volume 29, 2019, pp. 725–733.

[6] A. Kuestenmacher, N. Akhtar, P. G. Plöger, G. Lakemeyer, Towards robust task execution for domestic service robots, Journal of Intelligent & Robotic Systems 76 (2014) 5–33.

[7] K. Talamadupula, G. Briggs, M. Scheutz, S. Kambhampti, Architectural mechanisms for handling human instructions for open-world mixed-initiative team tasks and goals, Advances in Cognitive Systems 5 (2017) 37–56.

[8] P. Schermerhorn, M. Scheutz, Using logic to handle conflicts between system, component, and infrastructure goals in complex robotic architectures, in: 2010 IEEE International Conference on Robotics and Automation, 2010, pp. 392–397. doi:10.1109/ROBOT.2010.5509763.

[9] F. H. Van Eemeren, B. Garssen, E. C. Krabbe, A. F. S. Henkemans, B. Verheij, J. H. Wagemans, Handbook of Argumentation Theory, Dordrecht: Springer, 2014.

[10] L. Rizzo, L. Longo, An empirical evaluation of the inferential capacity of defeasible argumentation, non-monotonic fuzzy reasoning and expert systems, Expert Systems with Applications 147 (2020) 113220. URL: https://www.sciencedirect.com/science/article/pii/S0957417420300464. doi:https://doi.org/10.1016/j.eswa.2020.113220.

[11] A. Vassiliades, N. Bassiliades, T. Patkos, Argumentation and explainable artificial intelligence: a survey, The Knowledge Engineering Review 36 (2021).

[12] K. Atkinson, T. Bench-Capon, D. Bollegala, Explanation in AI and law: Past, present and future, Artificial Intelligence 289 (2020) 103387. URL: https://www.sciencedirect.com/science/article/pii/S0004370220301375. doi:https://doi.org/10.1016/j.artint.2020.103387.

[13] P. M. Dung, On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games, Artificial Intelligence 77 (1995) 321–357.

[14] L. Amgoud, C. Cayrol, M.-C. Lagasquie-Schiex, P. Livet, On bipolarity in argumentation frameworks, International Journal of Intelligent Systems 23 (2008) 1062–1093.

[15] O. Cocarascu, F. Toni, Argumentation for Machine Learning: A Survey, in: COMMA, 2016, pp. 219–230.

[16] M. Mozina, J. Zabkar, I. Bratko, Argument based machine learning, Artificial Intelligence 171 (2007) 922–937.

[17] P. Clark, T. Niblett, The CN2 Induction Algorithm, Machine Learning 3 (1989) 261–283.

[18] L. Amgoud, M. Serrurier, Agents that argue and explain classifications, Autonomous Agents and Multi-Agent Systems 16 (2008) 187–209.

[19] L. Carstens, F. Toni, Using argumentation to improve classification in natural language problems, ACM Transactions on Internet Technology (TOIT) 17 (2017) 30.

[20] N. Kotonya, F. Toni, Gradual argumentation evaluation for stance aggregation in automated fake news detection, in: Proceedings of the 6th Workshop on Argument Mining, Association for Computational Linguistics, Florence, Italy, 2019, pp. 156–166. URL: https://aclanthology.org/W19-4518. doi:10.18653/v1/W19-4518.

[21] M. Moens, Argumentation mining: How can a machine acquire common sense and world knowledge?, Argument and Computation 9 (2018) 1–14. doi:10.3233/AAC-170025.

[22] S. Eger, J. Daxenberger, I. Gurevych, Neural end-to-end learning for computational argumentation mining, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 11–22. URL: https://www.aclweb.org/anthology/

P17-1002. doi:10.18653/v1/P17-1002.
[23] M. Bishop, C. Gates, K. Levitt, Augmenting machine learning with argumentation, in: Proceedings of the New Security Paradigms Workshop, NSPW '18, ACM, New York, NY, USA, 2018, pp. 1–11. URL: http://doi.acm.org/10.1145/3285002.3285005. doi:10.1145/3285002.3285005.
[24] H. Ayoobi, M. Cao, R. Verbrugge, B. Verheij, Local-HDP: Interactive Open-ended 3D Object Category Recognition in Real-Time Robotic Scenarios, Robotics and Autonomous Systems (2021).
[25] A. Tarski, A lattice-theoretical fixpoint theorem and its applications., Pacific Journal of Mathematics 5 (1955) 285–309.
[26] A. Pazienza, S. Ferilli, F. Esposito, On the gradual acceptability of arguments in bipolar weighted argumentation frameworks with degrees of trust, in: International Symposium on Methodologies for Intelligent Systems, Springer, 2017, pp. 195–204.
[27] V. Losing, B. Hammer, H. Wersing, Incremental on-line learning: A review and comparison of state of the art algorithms, Neurocomputing 275 (2018) 1261–1274.
[28] G. Cauwenberghs, T. Poggio, Incremental and decremental support vector machine learning, in: Advances in Neural Information Processing Systems, 2001, pp. 409–415.
[29] A. Soula, K. Tbarki, R. Ksantini, S. B. Said, Z. Lachiri, A novel incremental Kernel Nonparametric SVM model (iKN-SVM) for data classification: An application to face detection, Engineering Applications of Artificial Intelligence 89 (2020) 103468. URL: https://www.sciencedirect.com/science/article/pii/S0952197619303501. doi:https://doi.org/10.1016/j.engappai.2019.103468.
[30] A. Bordes, S. Ertekin, J. Weston, L. Bottou, Fast kernel classifiers with online and active learning, Journal of Machine Learning Research 6 (2005) 1579–1619.
[31] A. Saffari, C. Leistner, J. Santner, M. Godec, H. Bischof, On-line random forests, in: Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 1393–1400.
[32] D. Dua, C. Graff, UCI machine learning repository, 2017. URL: http://archive.ics.uci.edu/ml.
[33] B. Gu, V. S. Sheng, K. Y. Tay, W. Romano, S. Li, Incremental support vector learning for ordinal regression, IEEE Transactions on Neural Networks and Learning Systems 7 (2015) 1403–1416.
[34] B. Gu, V. S. Sheng, Z. Wang, D. Ho, S. Osman, S. Li, Incremental learning for ν-support vector regression, Neural Networks 67 (2015) 140–150.
[35] J. R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.
[36] R. Agrawal, R. Bala, Incremental Bayesian classification for multivariate normal distribution data, Pattern Recognition Letters 29 (2008) 1873 – 1876. URL: http://www.sciencedirect.com/science/article/pii/S0167865508001992. doi:https://doi.org/10.1016/j.patrec.2008.06.010.
[37] P. Reiner, B. M. Wilamowski, Efficient incremental construction of RBF networks using quasi-gradient method, Neurocomputing 150 (2015) 349–356.
[38] J. Lu, F. Shen, J. Zhao, Using Self-Organizing Incremental Neural Network (SOINN) for radial basis function networks, in: 2014 International Joint Conference on Neural Networks (IJCNN), IEEE, 2014, pp. 2142–2148.

[39] H. Ayoobi, M. Rezaeian, Swift distance transformed belief propagation using a novel dynamic label pruning method, IET Image Processing 14 (2020) 1822–1831.
[40] B. Zhou, D. Bau, A. Oliva, A. Torralba, Interpreting deep visual representations via network dissection, IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (2019) 2131–2145. doi:10.1109/TPAMI.2018.2858759.
[41] H. Ayoobi, M. Cao, R. Verbrugge, B. Verheij, Argue to learn: Accelerated argumentation-based learning, in: 20th IEEE International Conference on Machine Learning and Applications (ICMLA), IEEE, 2021.