A new Transparent Ensemble Method based on Deep learning

Naziha Sendi, Nadia Abchiche-Mimouni, Farida Zehraoui

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2019) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

23rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

A new Transparent Ensemble Method based on Deep learning

Naziha Sendi[a,b], Nadia Abchiche-Mimouni[a,*], Farida Zehraoui[a,**]

[a] *IBISC, Univ Evry, Université Paris-Saclay, 91025, Evry, France*
[b] *VISIOMED group, 75016 Paris, France*

Abstract

Rather than making one model and hoping this model is the best/most accurate predictor we can make, ensemble methods which improve machine learning results by combining different models. However, one of the major criticisms is their being inexplicable, since they do not provide results explanation and do not allow prior knowledge integration. With the development of the machine learning the explanation of classification results and the ability to introduce domain knowledge inside the learned model have become a necessity. In this paper, we present a novel deep ensemble method based on argumentation that combines machine learning algorithms with multi-agent system to improve classification. The idea is to extract arguments from classifiers and to combine them using argumentation in order to exploit the internal knowledge of each classifiers and provide explanation behind decisions and to allow injecting prior knowledge. The results demonstrate that our method can effectively extract high quality knowledge for ensemble classifier and improve the performance.

Keywords: Machine learning, Deep learning, Extraction knowldge from neural networks, ensemble methods, argumentation...

1. Introduction

Deep learning (DL) models have started penetrating into critical areas like healthcare, justice systems and financial industry. The inability to explain (or interpret) DL models remains problematic in these areas. In order to be adopted, it is important to show that machine learning (ML) models base their predictions on a reliable representation of input data and do not focus on irrelevant artifacts that are present in the learning data. In addition, DL models are obtained from training large amount of data. This purely data-driven learning may induce contradictory results that can be uninterpretable. Injecting prior knowledge in deep neural networks (DNNs) is desirable to guide the learning step of models and reduce their uninterpretability. In ML, the essential part of explanability (or interpretatbility) is the ability to provide an intelligible explanation. In other words, ML results have to be understandable by humans [14]. Most

*Nadia Abchiche-Mimouni. Tel.: +3 316 485 3474 ; fax: +3 316 485 3601.
**Farida Zehraoui. Tel.: +3 316 485 3464 ; fax: +3 316 485 3601.
E-mail address: nadia.abchiche@ibisc.univ-evry.fr; farida.zehraoui@univ-evry.fr

of works in the literature propose to use interpretable known models to explain black boxes. Among these explained models we can cite: linear models, decision trees and logic rules [14]. In the present work, we focus on logic rules that provide a flexible way to express structured knowledge with textual representation. Furthermore, this representation "facilitates" the introduction of prior knowledge.
The most common rules form is the "$if - then$" one. Other types of rules like the "$m - of - n\ rules$" where given n conditions (premises), the conclusion (decision) is true if m conditions from n ($m < n$) are satisfied.
Rule extraction from ML algorithms is a hot research topic that has been widely explored and has shown significant contributions in the past. Over the last decades, many authors presented some techniques showing how to extract symbolic rules from a Neural Network (NN). The majority of the proposed approaches concern the NNs and few works address the problem of rule extraction from DNNs. This is due to the huge number of layers in these models and their complexity. In the 1990s, Andrews et al. [2] introduced a taxonomy aiming at characterizing rule extraction techniques. Essentially, rule extraction algorithms belong to three categories: decompositional, pedagogical, and eclectic. In decompositional techniques [27], [13], [32], rules are extracted using the internal structure of the NN models, at the level of hidden and output neurons by analyzing the weight values. This type of algorithms is specific to the NNs models and can not be applied to other classifier models. Pedagogical approaches [10], [3], [5] extract rules by using only the inputs and the outputs of NNs models. This kind of techniques is model agnostic and can be used for any classifier since it does not take into account the specifities of the classifier. The eclectic approaches [18], [22] consider elements of both decompositional and pedagogical techniques. The described approaches focus on extracting rules from a single NN. However, only few works explore Rule Extraction from Ensemble NNs [6],[31], [17].

In this work, we propose a novel DL-based ensemble method that rely on multiagent argumentation. Argumentation can be abstractly defined as the interaction of different arguments for and against some conclusion [25]. The idea is to provide arguments, extracted from DL classifiers by using the argumentation technique. Arguments are obtained from the rules extracted from each classifier of the ensemble. Each rule set extracted from one classifier is "associated to" an agent in a multiagent system (MAS).
In addition, the use of logic rules and a MAS has simplified the addition of prior knowledge. Indeed, it is enough to have a single agent that integrates prior knowledge, which are described by logic rules, to allow the argumentation system taking into account this kind of knowledge. So, in order to provide classification result, the decision is based both on the prior knowledge and the explanation of each output provided by the classifiers (using the extracted rules), whereas the classical ensemble methods are limited to the combination of the classifiers outputs.

The rest of this paper is organized as follows. The next section describes our method by detailing its different steps. Section 3 illustrates the proposed method by giving a case study. In Section 4, we carry out the experimental analysis over some public ML datasets. Finally, in Section 5, we make our concluding remarks and prospects.

2. Method

We propose an original and transparent Deep ensemble method that provides result explanation, integrates internal classification knowledge in base classifiers and allow prior knowledge. The model of our method consists of two phases: arguments extraction phase and multiagent argumentation phase (see figure 1).

2.1. Arguments extraction phase

First in sampling stage, individual agent generates its training sample by using a bootstrap sampling from the training data. As in bagging ensemble method [8], a bootstrap sample is obtained by a random selection of examples with replacement from the original training dataset.

2.1.1. Deep multilayer network description

We use a deep multilayer perceptron (DMLP) as base classifier for the ensemble method because it has shown good results in several fields. A DMLP consists of an input layer that receives input examples, hidden layers that are fully connected to the previous and the next layers and an output layer that provides the network outputs. These consist of the probabilities to belong to the classes. Let's h_i^l the i^{th} neuron of the hidden layer l, its activation is defined by: $h_i^l = f(\sum_j w_{ji}^l h_j^{l-1})$, where w_{ji}^l is the weight of the connection from the j^{th} neuron of the layer $(l-1)$ to the i^{th}

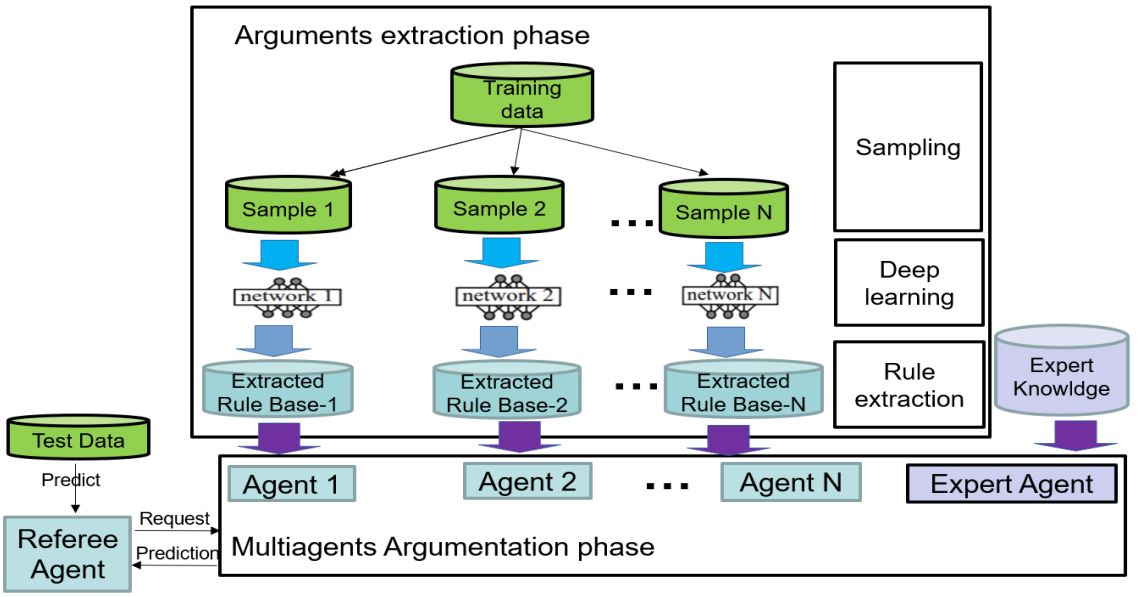

Fig. 1. Approach architecture

neuron of the layer l (h^0 represents the input layer) and f is the activation function. For the hidden layers, we used the Rectified Linear Units ($ReLU$) activation function, which gives good results in practice. It is defined as follows: $ReLU(x) = max(0, x)$. We used the softmax activation function for the output layer in order to obtain the probabilities that the input $X = (x_1, x_2, ..., x_n)$ belongs to a class c. This function is defined by: $softmax(h_c^o) = \frac{exp^{h_c^o}}{\sum_l exp^{h_l^o}}$
To train the DMLP, we used the adam [19] optimizer and the cross-entropy cost function L, which is the best choice in state-of-the art implementations. It is defined by: $L(Y, O) = -\frac{1}{N} \sum_i \sum_l y_{il} \ln(o_{il})$.

2.1.2. *Rules extraction step*

Despite their power of prediction, DNN are considered as black boxes, which makes their interpretation difficult. That's why it was necessary to characterize and interpret them. Rule extraction step allows to explain the predictions of classifiers. To extract classification rules from DNNs, we have evaluated one pedagogical approach [10] and one eclectic approach [5]. We have chosen these approaches because they are scalable and adapted to the use of multiple deep learning algorithms. In [10], the authors proposed an algorithm (TREPAN) that extracts rules from DNN by producing a decision tree. TREPAN uses a best-first procedure when building the decision tree. This consists in choosing the node that most increases the fidelity of the tree. A rule is defined by the path between the root and a leaf of the built tree. In [5], the first phase consists in approximating the discriminant frontier built by a DMPL using discriminant hyperplanes frontiers. In the second phase, a Discretized Interpretable Multilayer Perceptron (DIMLP) model is extracted based on the discriminant hyperplanes frontiers. The multilayer architecture of DIMLP is more constrained than that of a standard multilayer perceptron. From this particular constrained network, rules based on continuous attributes can be extracted in polynomial time.
The extracted classification rules from each classifier constitute a rule base that is associated to the classifier. A clasification rule corresponds to the weights of attributes however the path that led to these rules is different from one approach to another. In decompositional techniques, rules are extracted by analyzing weight values of the internal structure of NN. In Pedagogical approaches rules are extracted by using only the inputs and the outputs of NNs models. The form of a classification rule CR is: $CR : (pr_1)\ (pr_i)\ ...\ (pr_n) \Longrightarrow (class(CR) = c, confidence\_score(CR) = s)$, where: $pr_i \in premises(CR)$ $(1 \leq i \leq n)$ are the premises of the rule CR that the example must satisfy to be classified in $c \in C$ (C is the set of classes). The form of the premise pr_i is defined by $pr_i = (x_i\ op\ \alpha_i)$ where x_i is the value of the i^{th} attribute, α_i is a real number and op is an operator. s $(0 \leq s \leq 1)$ is a confidence score that is associated to the rule CR. This score depends on the number $ne_c^+(CR)$ of examples that are well classified by the rule CR. To take into

account the fact that most real datasets are unbalanced the number of well classified examples $ne_c^+(CR)$ is divided by the total number of examples ne_c in the class c : $confidence\_score(CR) = \frac{ne_c^+(CR)}{ne_c}$
Experts' domain knowledge is also modeled in the form of rules, named expert rules (ERs):
$ER : (pr_1)\ (pr_i)\ ...\ (pr_n) \Longrightarrow (class(ER) = c)$, where $pr_i \in premises(ER)$ $(1 \leq i \leq n)$ are the premises of the rule ER that the example must satisfy to be classified in the class $c \in C$ based on the official experts' knowledge. For example, the ER_1 rule below expresses that an official recommendation for hypertension is to prefer the beta blockers (BB) treatment for young people: $ER_1 : (age < 50) \Longrightarrow (class(ER_1) = BB)$.
Each rule base is encapsulated in an agent. In order to allow injecting prior knowledge in the system, an *Expert* agent is added for embedding the knowledge base which models prior knowledge provided by domain experts.

2.2. Multiagent argumentation phase

Argumentation theory is a branch of AI that studies reasoning with incomplete and conflicting information, one application of which is in the field of medical decision support systems. Our starting point is Dung's abstract argumentation theory [11]. Arguments are represented with a directed graph, where each node represents an argument and each arc denotes an attack by an argument on another. To express that an argument a attacks an argument b (that is, argument a is stronger than b and b is discarded), a binary relation is defined. The graph is analyzed to determine which set of arguments is acceptable according to general criteria. Structured argumentation has been introduced by [4] to formalize arguments in such a way that premises and claim (such as a class for a CR, see 2.1.2) of the argument are made explicit, and the relationship between premises and claim is formally expressed (for instance using rule deduction). In our case, the arguments need to be structured since they are already given in the form of rules. Several works propose semantics for determining acceptability of arguments. Preferences based approaches consider global evaluation of the arguments (extensions). Since in our distributed approach it is hard to use a global preference based argumentation, we exploited local (agent) preference based method. As we will see later, the score and the number of premises of the rules are used during the encounter of arguments. [20] distinguishes dialogical argumentation where several agents exchange arguments and counterarguments in order to argue for their opinion, from monological argumentation whose emphasis is on the analysis of set of arguments and counterarguments.

2.2.1. Modelling the argumentation process

In our approach, each agent of the MAS argues for its own prediction against other agents. So, we have focused on dialogical argumentation for the implementation of the argumentation process [26]. More precisely, agents engage in a process of persuasion dialogue [15] since they have to convince other agents that their prediction is better. Through the argumentation process, each agent uses the set of rules in its embedded rule base to answer to a prediction request and to provide arguments. Since all the agents are able to participate to the argumentation process by exchanging messages, we have focused on multilateral argumentative dialogues protocols [7]. According to [24], multilateral argumentative dialogue protocol (MADP) is based on several rules, that are instanciated in our approach as explained hereafter. Moreover, it has been shown in [1] that agents role affect positively the argumentation process. So, in order to organize the dialogue, four distinct agent roles are defined:
(i) Referee: agent who broadcasts the request for a prediction and manages the argumentation process;
(ii) Master: agent that answers first to the *Referee* request;
(iii) Challenger: agent who challenges the *Master* by providing arguments;
(iv) Spectator: agent who does not participate to the argumentation process.
The *Referee* is an "artifact" agent role that is assigned in a static way. This agent interacts with the user for acquiring the prediction request and collecting the final result. The argumentation process is performed through agents communication. For that purpose, we adopted speech acts language [28]. Let X be the input data, where X is a vector of attributes values $(x_i)_{i=1,...,n}$, c the class to predict, A_r the Referee Agent, A_e the Expert agent, A_m the agent whose role is Master and A_c the agent whose role is Challenger. Seven communication performatives are used to instanciate the rules of the MADP as follows:

1. **Starting rules**: A_r uses the REQUEST performative to brodcast the request for a prediction. The content of the message is: $(X, ?c)$.

2. **Locution rules**: an agent A_i sends an information by using the INFORM performative and asks for an information by using the ASK performative.
3. **Commitment rules**: two rules are defined. The first one is for managing the request for a prediction by using the PROPOSE performative. This performative allows an agent A_i to propose an opinion by selecting the best rule that matches the request: $R_x^{i*} \in RB_x^i$ such that $confidence\_score(R_x^{i*}) = \max_{R_x^i \in RB_x^i} (confidence\_score(R_x^i))$, where $RB_x^i = \left\{R^i : R^i \in RB^i \wedge premises(R^i) \subset x\right\}$ (RB^i is the rule base associated to the agent A_i).
   The second rule allows an agent to declare its defeat by using the DEFEAT performative.
4. **Rules for combination of commitments:** three rules for dealing with COUNTER, DISTINGUISH, BE_INAPLICABLE performatives are defined. They define how acceptance or rejection of a given argument is performed.
   COUNTER: A_c uses this speech act to attack the argument of A_m (associated to the rule R_x^{m*}) by selecting the rule R_x^{c*} such that $confidence\_score(R_x^{c*}) > confidence\_score(R_x^{m*})$.
   DISTINGUISH: A_c uses this speech act to attack the opponent's argument, in case of equality of rule scores of A_c and A_m, they use the number of premises in their proposed rules as arguments: If $premise\_number(R_x^{c*}) > premises\_number(R_x^{m*})$ then A_c becomes the new Master ($premises\_number$ is the number premises of a rule).
   BE_INAPLICABLE: The expert agent A_e uses this speech act to check if the proposed rule R_x^{i*} by an agent A_i does not violate the rules $R_x^* e \in RB^e$.
5. **Termination rules**: the dialogue ends when no agent has a rule to trigger.

2.2.2. Agents dialogues specification and behavior

At the beginning, Referee broadcasts the discussion topic,by using the performative REQUEST(X,?c). This means that the goal is to predict the class of the vector X. Each agent produces an opinion by selecting the best rule that matches the request and uses the performative PROPOSE to send it to the Referee. Evry agent who have an opinion about the current topic, sends it to the Referee agent which in turn sends the proposed opinion to the Expert agent for verification using the performative INFORM. Expert agent verify if the opinion matches with the recommendations, if there is no conflict with the expert knowledge, then he sends a message to the Referee agent by using the performative BE_Inapplicable to express his acceptance else he sends a rejection. The first agent who proposes an accepted opinion about the current topic will be selected as the Master. All the other participants can challenge the Master and form the queue of challengers, and the first participant in the queue is selected to be the Challenger. All the other participant agents except Master and Challenger are Spectators. Once Master and Challenger are identified, agents can use the speech acts for constructing Master-Challenger dialogues. The Challenger ask the Master for his arguments by using ASK performative. Moreover he can use performatives COUNTER and DISTINGUISH to attack Master arguments. If Master is defeated by Challenger, he uses the performative DEFEAT and this Challenger will become the new Master, and he can propose his opinion about the current topic from his own knowledge base. All the other participants decide whether or not to challenge this new opinion. Noted that the defeated argument of the old Master can't be used again, the old Master can only produce a new argument to apply for Master once more. Otherwise, if Challenger is defeated, the next participant in the challenger queue will be selected as the new Challenger, and the argumentation continues. Argumentation stops when there is no agent applying for Master. Since the number of arguments produced by participants is finite and the defeated arguments can't be allowed to use repeatedly, the termination of argumentation can be guaranteed. The final prediction belongs to the agent who resist attacks in argumentation process and whose argument is the most robust.

3. Case study

In order to illustrate the argumentation process and show the relevance of our approach, we propose a simple case study based on a specific dataset that is a realistic virtual population [23] with the same age, sex and cardiovascular risk factors profile than the French population aged between 35 and 64 years old. It is based on official French demographic statistics and summarized data from representative observational studies. Moreover, a temporal list of visits is associated to each individual. For the current experiments, we have considered 40000 individuals monitored for hypertension during 10 visits per individual. Each visit contains: the systolic blood pressure (SBP), diastolic blood

pressure (DBP), etc. For hypertension treatment, 6 major classes of drugs have been considered: calcium antagonist (AC), beta blockers (BB), ACE inhibitors (IEC), diuritics (DI), sartans (SAR) and No treatment (NN). The data have been used to predict the optimal treatment based, following the steps described in the previous section.

3.1. Scenario illustration

In this scenario, 5 agents, including the Referee agent, are involved in the argumentation process. The goal is to predict a treatment for a given patient X. The first step of our approach consists in building diverse DMLP models from bootstrap samples. Before the rule extraction step, we evaluated the diversity of the DNNs. We defined the diversity between two DMLPs as the percentage of disagreements between the classifier outputs. Other measures for computing the diversity can be found in [21]. Table 1 shows the diversity scores between every couple of DMLPs. We can see that in average, the diversity score is around $19,2\%$. The score is not satisfying but this is just an illustrative simple example. In practice, we select the more diverse classifiers by defining a threshold.

Table 1. Diversity of Dimlps.

	$DMLP^1$ -$DMLP^2$	$DMLP^1$-$DMLP^3$	$DMLP^1$-$DMLP^4$	$DMLP^2$-$DMLP^3$	$DMLP^2$- $DMLP^4$	$DMLP^3$-$DMLP^4$
Diversity	0.255	0.064	0.183	0.311	0.214	0.123

In rule extraction phase, we have extracted knowledge bases from DNNs using the eclectic rule extraction approach proposed in [5]. Table 2 shows the properties of the four rule bases extracted from the four DMLPs in terms of number of rules per base, examples per rule, premises and premises per rule. Each rule base is then embedded in

Table 2. Rule bases properties.

Rule Bases propperties	RB^1	RB^2	RB^3	RB^4
Number of rules	159	246	197	267
Number of premises	75	83	49	38
Number of premises per rule	6.7	8.9	9.5	8.7
Number of examples per rule	420.4	352.3	652.7	201.4

an agent. The rules are thus considered as individual knowledge of the agents. The possible negotiation arguments are the $confidence\_score$ of the rules and their premisses number. In the beginning, Referee agent broadcasts the request, by sending the vector of the patient X and the class to predict. In addition, we injected few medical recommendations for hypertension treatment into the expert agent A_e. Examples of these recommendations: R^e_1: (age > 50 years) $\Longrightarrow$ $(class(R^e_1) = BB)$) and R^e_2: (age < 50 years) $\Longrightarrow$ $(class(R^e_2) = DI)$)

The execution of the argumentation process is illustrated using the patient example p:$[age = 51][sex = female][Visit_0 : SBP = 100.2,\ DBP = 81.3][Visit_1 : SBP = 135.9,\ DBP = 99.0][Visit_2 : SBP = 113.6,\ DBP = 76.9][Visit_3 : SBP = 112.1,\ DBP = 77.2]$.

p should be treated by the treatment BB and the objective of the system is to predict this optimal treatment following the argumentation process illustrated in Figure 2. At the first iteration T_1, the Referee agent broadcasts the prediction request by transmitting the attributes p and the requested class $?c$ to predict. Each agent produces his opinion by selecting the best rule that matches the request. At T_2 Agent 2 becomes the first Master that proposes his opinion as follows: "the requested class should be BB based on the rule: R^{2*}_p: $(age > 50)(SBP_{Visit1} > 99.0) \Longrightarrow (class(R^{2*}_p) = BB,\ confidence\_score(R^{2*}_p) = 0.6)$". At T_3, the Referee Agent sends the proposed opinion of Agent 2 to Expert Agent for verification who checks if the opinion matches with the recommendations. He (Expert Agent) then sends a message to the Referee Agent to express its acceptance since there is no conflict to declare. At T_4, Referee Agent declares that agent 2 is defined as a Master. At T_5, Expert Agent proposes his opinion as follows: "the requested class should be DI based on the rule: R^{3*}_p: $(age > 50)(DBP_{Visit2} > 72.1)(DBP_{Visit3} > 77.1) \Longrightarrow (class(R^{3*}_p) = DI,\ confidence\_score(R^{3*}_p) = 0.5)$". In this case, Expert Agent declares that $confidence\_score(R^{2*}_p)$ is

inapplicable since the predicted class *DI* (given by this rule) does not match with the predicted class of the recommendation rule: R_1^e: (age > 50 years) $\Longrightarrow$ (*BB*, 1), at T_6. At T_7, Agent 4 proposes his opinion as follows:"the requested class should be *BB* based on the rule: R_p^{4*}: $(age > 50)(DBP_{Visit0} > 80.1)(SBP_{Visit1} > 129.1)(SBP_{Visit3} > 100.7) \Longrightarrow (class(R_p^{4*}) = BB,\ confidence\_score(R_p^{4*}) = 0.6)$". At T_8, A_e declares that this rule is applicable since there is no conflict. At T_9, Referee Agent declares that Agent 4 is the first Challenger, Agent 1 and Agent 3 are Spectators. Since a Master and a Challenger are defined, the encounter arguments can be performed. At T_{10}, Agent 4 (Challenger) asks Agent 2 (Master) for his arguments in order to compare them with his own arguments. At T_{11}, Agent 2 sends his arguments to Agent 4. In this case, the score of the rule R_p^{4*} (Agent 4) is equal to the score of R_p^{2*} (Agent 2). In such a situation, the number of premises of the two rules are compared and it is found that the number of premises of rule R_p^{2*} ($npr(R_p^{2*})$) is lower than the number of premises of rule R_p^{4*} at T_{12}. Thus, Agent 2 admits his defeat and Agent 4 becomes the new Master and can propose its own opinion at T_{13}. The argumentation process continues until none of the agents is able to propose an opinion nor challenging another agent opinion. At the end, the final master gives his prediction of the hypertension medication in the form of a rule which is easy to understand. The patient *p* has been well classified and the system recommends him to take *BB* treatment based on the rule of Agent A_1: R_p^{1*}: $(age > 50)(DBP_{Visit0} > 79.3)(SBP_{Visit0} > 100.0)(SBP_{Visit3} > 101.5) \Longrightarrow (class(R_p^{1*}) = BB,\ confidence\_score(R_p^{1*}) = 0.7)$.

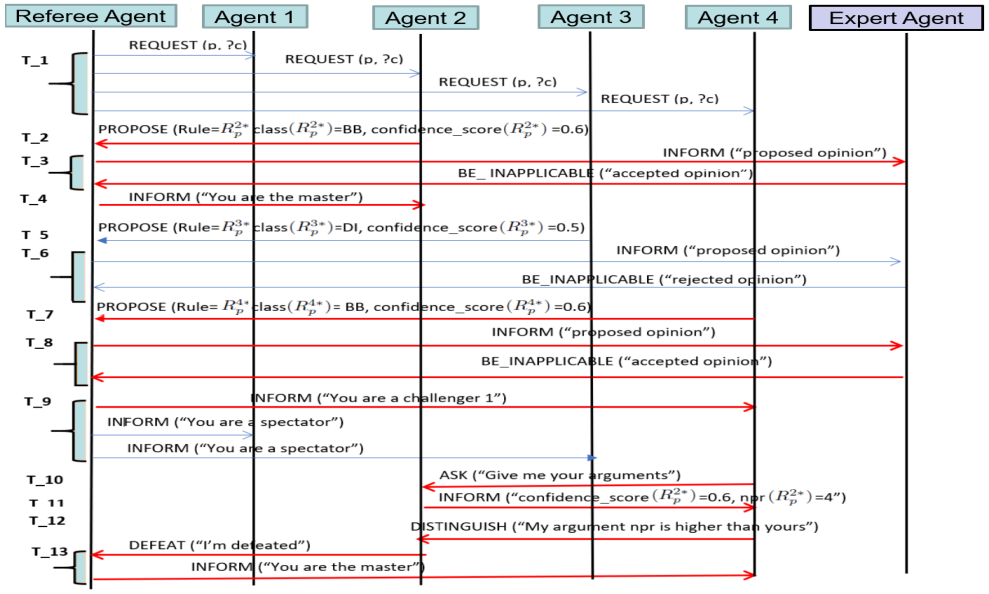

Fig. 2. Illustration of the case study argumentation process

The evaluation of our approach on the hole dataset using cross validation shows that our approach, without using prior knowledge gives an accuracy of about 83.2% and 89% after the knowledge injection while we obtain an accuracy of 79.8% with a single DMLP. As a conclusion, we can confirm that our method adds a reasoning aspect to ensemble methods with the integration of argumentation in the way of combining DIMLPs. Thus, The arguments are extracted automatically from classifiers. Moreover, not only the class of the instance is given, but also the reasons behind that classification are provided to the user using comprehensible classification rules. Indeed, the system gives the path that led to the prediction. In Figure 2, red bold arrows are the most important messages that led to a final prediction treatment for patient *p*. In addition, our approach can deal with the injection of expert knowledge wich contributes to the transparency of the system by controlling the prediction process.

4. Experimentation

In the experiments we use 11 public UCI datasets (https://archive.ics.uci.edu/ml/datasets.html [53]) representing classification problems of two classes. Our experiments are based on 10 repetitions of 10-fold cross-validation trials. Training sets were normalized using Gaussian normalization. We have tested three variants of our approach: (i) App1_DIMLP that uses the electic rule extraction algorithm described in [5]; (ii) App2_TREPAN that uses the pedagogical rule extraction algorithm described in [9] and (iii) App3_Ext that replaces the DMLPs and the rule extraction step by a rule extraction algorithm that extracts rules directly from the bootstrap samples.
In order to validate the performance of our approach, we compared the three variants described above to: (i) the most popular ensemble learning methods (Bagging [8], AdaBoost [12]) and (ii) two classification approaches based on ensemble rule extraction that uses the DIMPL [6]: one trained by bagging (DIMLP-B) and another trained by arcing (DIMLP-A). We defined a grid search to optimize the parameters of each approach. The number of the bootstrap samples used in all the approaches is shown in Table 3. For DIMLP ensembles, we have used the default parameters defined in [6](for example, the number of bootstrap samples is equal to 25). In the argumentation process, we have only used *CRs* because there are no *ERs* provided for the used public datasets. Therefore the argumentation process takes place without any expert agent. In the experiment, we have used the Accuracy and Fidelity and Diversity as evaluation measures to compare the classification performance of the different methods described above. Accuracy indicates the percentage of well-predicted data and Fidelity indicates the degree of matching between network classifications and rules' classifications. The Diversity represents the average of the percentage of output disagreements between each couple of classifiers.

Table 3 that the diversity of our approach is greater than 30% for all the datasets. This confirm the fact that we obtain diverse classifiers from bootstrap sampling. This allows our method to reduce the error compared to a single DMLP. Table 3 shows also that our framework can effectively ensure high accuracy results for the classification task on several datasets.
In contrast with traditional ensemble methods, we can find that App1_DIMLP and App2_TREPAN outperform Bagging and AdaBoost methods using fewer classifiers. For example in Vertebral Column dataset, App1_DIMLP obtains an accuracy of 86.8% (using 11 classifiers) while the accuracy of Bagging and AdaBoost are lower than 75% (using more than 125 classifiers). Our method gives better results than DIMLP-B and DIMLP-A on the majority of datasets. For example, in Bupa Liver Desorders dataset, the accuracy of App1_DIMLP exceeds that of DIMLP-A by 25.2%. In Breast Cancer Prognostic dataset, the accuracy of App1_DIMLP is 88.7% (using 10 classifiers) while the accuracies of DIMLP-B and DIMLP-A are lower than 75%. So far, the results have been in our favor for the predictive accuracy of the 10 out of 11 classification problems. Moreover, we can see that the Fidelity score is higher than 95% in all datasets. This means that the classification rules extracted from the DMLPs matches the classification results provided by the DMLPs.
Experimental results show that App1_DIMLP and App2_TREPAN give better accuracy for the classification task than other ensemble methods. Indeed the use of argumentation process allows to outperform the classical ensemble methods and also the rules extracted from ensembles. In addition, we have shown that using deep learning with rule extraction step gives better results than using a rule extraction algorithm directly from the bootstrap samples (App3_Extract). As a conclusion, we can deduce that App1_DIMLP and App2_TREPAN can effectively extract high quality knowledge for ensemble classifier and ensure high accuracy in classification as well. Moreover our method provides explanations and transparency of the predictions.

4.1. Discussion

Very few works have previousely adressed rule extraction from ensembles. The DIMLP was used to extract from network ensembles (DIMLP-A and DIMLP-B) [6]. Zhou et al. proposed the REFNE algorithm (Rule Extraction from Neural Network Ensemble) [31], that extracts rules from instances generated from the trained ensembles. Hayashi and his co-authors have extended the "Recursive-Rule eXtraction" (Re-RX) algorithm to multiple MLP Ensemble [17]. None of these works used argumentation for performing predictions nor addressed the problem of knowledge injection into the algorithm. Over the last decades, argumentation has come to be increasingly central as a core study within AI since it attracts much attention in a lot of fields, especially in ML. Existing approaches achieve different and desirable outcomes, ranging from improving performances (reduce the combinatory search among possible hypothe-

Table 3. Results comparison to ensemble methods.

Datasets	Adaboost	Bagging	DIMLP-B	DIMLP-A	App3_Ext	App1_DIMLP		App2_TREPAN		Single DMLP	Av.Diversity
	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	Fidelity	Accuracy	Fidelity		
Breast Cancer Prognastic	82.5±0.05 (150)	79.1±0.03 (125)	74.4±0.02 (25)	73.7±0.04 (25)	71.1±0.01 (12)	**88.7**±0.03 (10)	98.8±0.03	84.3±0.07 (11)	97.9±0.09	75.2±0.02	0.399
Bupa Liver Disorders	83.2±0.09 (125)	78.0±0.21 (100)	67.3±0.08 (25)	61.9±0.03 (25)	75.8±0.03 (11)	**87.1**±0.02 (10)	96.6±0.01	83.6±0.10 (10)	97.3±0.08	81.03 ±0.01	0.311
Glass	81.5±0.04 (100)	79.0±0.03 (100)	74.1±0.03 (25)	**81.9**±0.06 (25)	76.9±0.08 (12)	79.9±0.10 (11)	97.5±0.01	81.3±0.06 (10)	96.3±0.06	69.9±0.07	0.483
Haberman	74.6±0.05 (100)	72.0±0.01 (125)	76.4±0.08 (25)	74.3±0.09 (25)	72.4±0.04 (12)	81.4±0.02 (10)	97.8±0.03	**83.7**±0.05 (10)	97.9±0.01	72.0±0.06	0.51
Heart Disease	86.3±0.06 (100)	86.0±0.09 (100)	84.9±0.05 (25)	81.3±0.07 (25)	83.1±0.10 (12)	**86.6**±0.01 (25)	97.1±0.01	77.1±0.03 (11)	97.0±0.03	77.6±0.09	0,438
ILPD (Liver)	73.4±0.09 (150)	71.1±0.01 (125)	69.3±0.02 (25)	70.2±0.05 (25)	70.0±0.06 (12)	**79.1**±0.01 (11)	96.9±0.01	74.9±0.03 (10)	95.8±0.07	68.5±0.01	0,539
Pima Indians	78.1±0.09 (100)	77.8±0.06 (100)	77.4±0.06 (25)	76.1±0.04 (25)	77.2±0.01 (12)	**80.9**±0.02 (9)	97.8±0.07	77.6±0.01 (12)	96.9±0.01	79.1±0.07	0,448
Saheart	72.1±0.11 (150)	72.3±0.12 (100)	72.3±0.02 (25)	70.6±0.04 (25)	71.3±0.02 (9)	**74.8**±0.09 (11)	97.1±0.03	72.1±0.13 (12)	95.7±0.03	66.5±0.01	0,357
Sonar	72.4±0.01 (100)	70.6±0.01 (100)	71.1±0.06 (25)	74.3±0.06 (25)	71.0±0.05 (11)	76.6±0.04 (10)	96.9±0.01	**79.9**±0.06 (9)	96.7±0.06	59.9±0.04	0,546
Spect Heart	71.9±0.03 (125)	72.3±0.06 (150)	72.9±0.01 (25)	70.9±0.02 (25)	72.9±0.02 (11)	79.7±0.02 (9)	96.9±0.01	**81.9**±0.01 (12)	96.7±0.03	69.4±0.01	0.308
Vertebral Column	74.9±0.03 (125)	72.3±0.01 (150)	82.9±0.03 (25)	81.1±0.05 (25)	80.6±0.03 (12)	**86.8**±0.04 (11)	96.9±0.02	77.1±0.03 (10)	95.9±0.02	58.3±0.02	0,447

ses) to rendering the ML process more transparent by improving its explanatory power. All of these works illustrate the importance of building arguments for explaining ML examples. But all of them are dedicated to rule association [29] [30] or to decision trees [16]. Since the existing approaches are built in a monolitic way (i.e. based on a monolitic algorithm), they lack robustness. If the algorithm fails, the whole system fails. In contrast, our approach consists in distributing the argumentation process through agents where embedded rule bases act in autonomic way while argumenting with each other. Finally, the most important point is that, none of the existing approaches that combine ML and argumentation addresses deep learning methods, despite their power of prediction. In contrast with tradiotional ensemble method, our method can provide result explanation and integrates internal classification knowledge in base classifiers rather than only classification results.

5. Conclusion

In this paper, we have proposed a transparent deep ensemble method based on multiagent argumentation for classification. We have used argumentation to combine deep learning algorithms. Experiments show that as ensemble method, our approach significantly outperforms usual ensemble methods. In addition, our method effectively provides explanation behind decisions and therefore addresses the recent need for Explainable AI. The explanation provided

to the user is easy to grasp so one will be able to judge the acceptance of decisions. Moreover, it's easy to add agents who contains prior domain knowledge. The prospects of this work are various. In the short term, it will be necessary to carry out experiments on a larger scale in order to consolidate the results of our approach with the real electronic health record to realize several tasks such as diagnosis, prediction of the next visit date, etc. We also plan to diversify learning algorithms by testing the whole system using other algorithms such as deep recurrent neural networks, convolutional neural networks, etc. We will diversify negotiation protocol in order to improve the interaction process.

References

[1] Amgoud, L., Parsons, S., Maudet, N., 2000. Arguments, dialogue, and negotiation, in: ECAI.

[2] Andrews, R., Diederich, J., Tickle, A.B., 1995. Survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge-Based Systems 8, 373 – 389. Knowledge-based neural networks.

[3] Augasta, M.G., Kathirvalavakumar, T., 2012. Reverse engineering the neural networks for rule extraction in classification problems. Neural Processing Letters 35, 131–150.

[4] Besnard, P., Garcia, A., Hunter, A., Modgil, S., Prakken, H., Simari, G., Toni, F., 2014. Introduction to structured argumentation. Argument & Computation 5, 1–4.

[5] Bologna, G., Hayashi, Y., 2016. A rule extraction study on a neural network trained by deep learning, in: 2016 International Joint Conference on Neural Networks, IJCNN 2016, Vancouver, BC, Canada, July 24-29, 2016, IEEE. pp. 668–675.

[6] Bologna, G., Hayashi, Y., 2018. A comparison study on rule extraction from neural network ensembles, boosted shallow trees, and svms 2018, 1–20.

[7] Bonzon, E., Maudet, N., 2012. On the outcomes of multiparty persuasion, in: Proceedings of the 8th International Conference on Argumentation in Multi-Agent Systems, Springer-Verlag, Berlin, Heidelberg. pp. 86–101.

[8] Breiman, L., Breiman, L., 1996. Bagging predictors, in: Machine Learning, pp. 123–140.

[9] Craven, M., Shavlik, J.W., 1994. Using sampling and queries to extract rules from trained neural networks, in: ICML.

[10] Craven, M.W., Shavlik, J.W., 1995. Extracting tree-structured representations of trained networks, in: Proceedings of the 8th International Conference on Neural Information Processing Systems, MIT Press, Cambridge, MA, USA. pp. 24–30.

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

[12] Freund, Y., Schapire, R.E., 1996. Experiments with a new boosting algorithm.

[13] Fu, L., 1994. Rule generation from neural networks 24, 1114 – 1124.

[14] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., Pedreschi, D., 2018. A survey of methods for explaining black box models. ACM Comput. Surv. 51, 93:1–93:42.

[15] H., P., 2009. Models of Persuasion Dialogue. Simari G., Rahwan I. (eds) Argumentation in Artificial Intelligence. Springer, Boston, MA. pp. 34–41.

[16] Hao, Z., Yao, L., Liu, B., Wang, Y., 2014. Arguing Prism: An Argumentation Based Approach for Collaborative Classification in Distributed Environments. Springer International Publishing, Cham. pp. 34–41.

[17] Hayashi, Y., Fujisawa, S., 2015. Strategic approach for multiple-mlp ensemble re-rx algorithm, in: International Joint Conference on Neural Networks (IJCNN'2015), pp. 1–8.

[18] Hruschka, E.R., Ebecken, N.F., 2006. Extracting rules from multilayer perceptrons in classification problems: A clustering-based approach. Neurocomputing 70, 384 – 397. Neural Networks.

[19] Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. CoRR abs/1412.6980.

[20] Kontarinis, D., 2014. Debate in a multi-agent system : multiparty argumentation protocols.

[21] Kuncheva, L.I., Whitaker, C.J., 2003. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Mach. Learn. 51, 181–207. URL: https://doi.org/10.1023/A:1022859003006, doi:10.1023/A:1022859003006.

[22] Lu, H., Setiono, R., Liu, H., 1996. Effective data mining using neural networks. IEEE Trans. on Knowl. and Data Eng. 8, 957–961.

[23] Marchant, I., Hanane, E.M., Kassaí, B., Bejan-Angoulvant, T., Massol, J., Vidal, C., Amsallem, E., Naudin, F., Galan, P., Czernichow, S., Nony, P., Gueyffier, F., 2009. Score should be preferred to framingham to predict cardiovascular death in french population 16, 609–15.

[24] Mcburney, P., Parsons, S., 2002. Dialogue games in multi-agent systems. Informal Logic 22, 2002.

[25] Rahwan, I., Simari, G.R., 2009. Argumentation in Artificial Intelligence. 1st ed., Springer Publishing Company, Incorporated.

[26] Reed, C., 2006. Representing dialogic argumentation. Knowledge-Based Systems 19, 22–31.

[27] Sato, M., Tsukimoto, H., 2001. Rule extraction from neural networks via decision tree induction, in: IJCNN'01, pp. 1870 – 1875 vol.3.

[28] Searle, J., 1969. Speech acts. an essay in the philosophy of language. Cambridge University Press.

[29] Wardeh, M., Coenen, F., Bench-Capon, T., 2012. Multi-agent based classification using argumentation from experience. Autonomous Agents and Multi-Agent Systems 25, 447–474.

[30] Xu, J., Yao, L., Li, L., 2015. Argumentation based joint learning: A novel ensemble learning approach. PLOS ONE 10, 1–21.

[31] Zhou, Z.H., Jiang, Y., Chen, S.F., 2003. Extracting symbolic rules from trained neural network ensembles. AI Commun. 16, 3–15.

[32] Zilke, J.R., Loza Mencía, E., Janssen, F., 2016. Deepred – rule extraction from deep neural networks, in: Calders, T., Ceci, M., Malerba, D. (Eds.), Discovery Science, Springer International Publishing, Cham. pp. 457–473.