Approximate Classification with Web Ontologies through Evidential Terminological Trees and Forests

Giuseppe Rizzo*, Nicola Fanizzi**, Claudia d'Amato**, Floriana Esposito

LACAM – Dipartimento di Informatica — Università degli Studi di Bari "Aldo Moro", Campus Universitario, Via Orabona 4, 70125 Bari, Italy

Abstract

In the context of the Semantic Web, assigning individuals to their respective classes is a fundamental reasoning service. It has been shown that, when purely deductive reasoning falls short, this problem can be solved as a prediction task to be accomplished through inductive classification models built upon the statistical evidence elicited from ontological knowledge bases. However also these data-driven alternative classification models may turn out to be inadequate when instances are unevenly distributed over the various targeted classes To cope with this issue, a framework based on logic decision trees and ensemble learning is proposed. The new models integrate the *Dempster-Shafer theory* with learning methods for *terminological decision trees* and *forests*. These enhanced classification models allow to explicitly take into account the underlying uncertainty due to the variety of branches to be followed up to classification leaves (in the context of a single tree) and/or to the different trees within the ensemble model (the forest). In this extended paper, we propose revised versions of the algorithms for learning *Evidential Terminological Decision Trees* and *Random Forests* considering alternative heuristics and additional evidence combination rules with respect to our former preliminary works. A comprehensive and comparative empirical evaluation proves the effectiveness and stability

*Principal corresponding author

**Corresponding author

Email addresses: giuseppe.rizzo1@uniba.it (Giuseppe Rizzo), nicola.fanizzi@uniba.it (Nicola Fanizzi), claudia.damato@uniba.it (Claudia d'Amato), floriana.esposito@uniba.it (Floriana Esposito)

of the classification models, especially in the form of ensembles.

Keywords: ontologies, logic decision trees, Dempster-Shafer theory, instance classification

1. Introduction

Sharing knowledge that is encoded along formal ontologies, thus enabling rich reasoning capabilities, plays a key role in the context of the *Semantic Web* (SW). However, standard deductive inference mechanisms sometimes show their limitations because of the inherent incompleteness of the ontological knowledge bases combined with the adoption of an open-world semantics, which is natural in such a Web-scale heterogeneous and distributed context.

In order to tackle the consequences of these distinctive aspects, alternative forms of reasoning, based on statistical models that can be induced through data-driven methods, have been introduced for performing various tasks such as *concept retrieval* and *query answering* [1] more effectively. It has been shown that these tasks have been cast as *classification* problems, which amount to deciding the membership of an individual with respect to a target concept, and they have been solved through inductive learning methods exploiting statistical regularities in the underlying knowledge base. Specifically, the resulting models have been used by approximate classification procedures applied to the knowledge bases also in combination with deductive inference services [2]. The application of these methods has shown interesting results such as the ability to synthesize new concepts and/or produce inductive classification models inspired by *Inductive Logic Programming* (ILP) like *terminological decision trees* [3], i.e. *logic decision trees* [4, 5] whose inner node tests are expressed in terminological languages (that is Description Logics [6]). Additionally, exploiting such statistical models, non logically-derivable yet still consistent assertional knowledge may be suggested.

However, such alternative methods and models have also revealed some shortcomings. One of the issues is that they do not allow an explicit representation of uncertainty to be specifically exploited for managing those cases when the classification procedure assigns an uncertain membership. To better tackle these cases, an enhanced model, called *evidential terminological decision tree* has been devised, by integrating primitives of the *Dempster-Shafer Theory* [7]. The main advance with respect to terminological decision trees regards the heuristic used to select the concept installed into inner nodes (based on the *non-specificity measure* [8] rather than the classic measures stemming from *information gain*) and the classification procedure (that explores all the possible paths departing from a node with an uncertain test result).

Another issue concerns the distribution of the training data. In general, the individuals that are known (or can be logically assessed as) positive and negative instances for a given target concept (that is those that are instances of a target concept or of the negated target concept) may not be equally distributed. This skewness may be noticeably larger when considering individuals whose membership cannot be assessed by reasoning under an open-world semantics. This class-imbalanced setting may affect the model, resulting in poor performances. Various methods have been devised to tackle the general unbalance learning problem (see [9] for a survey of the various approaches). As regards the specific task of learning instance classification models for inductive query answering on SW knowledge bases, we investigated the adoption of methods for *ensemble models* [10] that are made up of a certain number of classifiers, trained by the so-called *weak learners*, and whose final prediction results from the combination of the predictions made by each classifier. Specifically, the combination is given by a specific rule playing the role of the *meta-learner*. Particularly, we proposed an algorithm for inducing *terminological random forests* [10] that extends *(First Order) random forests* [11, 12] with the use of Description Logics: the model is an ensemble of terminological decision trees [3].

Employing these models, the membership of a test individual w.r.t. a target concept is decided according to a majority vote rule (although various other strategies for combining predictions have been proposed [13, 14, 15]): each classifier equally contributes to the final decision returning a vote in favor of a single membership. In this way, some other aspects are not considered explicitly, such as the uncertainty about the single membership-label assignments and the disagreement that may intervene among weak learners. Particularly, the latter issue is crucial for the performance of ensemble models [16]: using the aforementioned type of forests, we noted that most misclassification cases were related to situations in which votes are evenly distributed with respect to the admissible labels. A weighted voting procedure may be an alternative strategy to mitigate the problem, but it requires a criterion for setting the weights.

In this sense, introducing a meta-learner which can manipulate the *soft* predictions made by each classifier (i.e. a prediction with a confidence measure for each membership value) rather than *hard predictions* (where only the predicted label is returned) may be a solution. Adopting the random forests as ensembles, this can be accomplished by considering evidential terminological decision trees [7] as base models. Dempster-Shafer theory has already been used in combination with ensemble learning procedures (e.g. see [17]). However, most of the methods apply to problems that involve simpler knowledge representations. Additionally, none of them has been employed for predicting assertions on ontological knowledge bases.

Therefore, we further extended the model proposing a framework for the induction of *Evidential Terminological Random Forests* for ontological knowledge bases [18]. Employing evidential terminological decision trees, the approach does not require the computation of decision templates. After the induction of the forest, new individuals are classified by combining the evidence on the membership prediction made by each tree through Dempster's rule [19].

However, we noted that the proposed framework had some limitations [7, 18]. Firstly, the heuristic to select the most promising label adopted by evidential terminological decision tree learning algorithm did not consider the presence of conflicting evidence. Secondly, the combination rule represented a bottleneck of the classification step: therefore it is important to investigate alternative solutions for improving the efficiency of the classification. Thirdly, the size of evidential terminological random forests seemed not to affect the predictiveness of the ensemble model (due to a weak diversification of the ensemble) but represented a source of complexity during the classification step.

Consequently, in this paper we extended the framework for learning evidential terminological decision trees and random forests along the following directions:

- we used different heuristics based on other total uncertainty measures (than the sole non-specificity measure) to drive the selection of the concepts to be installed into the nodes of evidential terminological decision trees;
- we used further combination rules to pool the evidence obtained by traversing each tree;
- we used further combination rules as meta-learner for evidential terminological random forests;
- we set up a comprehensive and comparative experimental evaluation showing the effectiveness of the proposed extensions when performing inductive instance retrieval.

The remainder of the paper is organized as follows: the next section introduces basics on the targeted representation language and the problem we aim to solve, that is inducing classifiers for the SW context; Sect. 3 recalls the basics on Dempster-Shafer Theory, required for understanding the framework of the evidential tree-based models presented in Sect. 4. In Sect. 5, the empirical evaluation of the classification models is described, while Sect. 6 discusses related approaches. Sect. 7 draws conclusions and illustrates some perspectives for further developments.

2. Basics of Description Logics and Problem Definition

In this section we recall the basics of *Description Logics* (DLs), that is the family of knowledge representation languages at the core of the standard *Web* *ontology language*[1] (OWL - DL).

In DLs, a domain is modeled in terms of a set of *atomic concepts*, $N_C = \{A, B, \cdots\}$ and *atomic roles*, $N_R = \{R, S, \cdots\}$. Two noteworthy concepts are the *top concept*, denoted with $\top$, and the *bottom concept*, denoted with $\bot$. DLs are endowed with a set of operators to combine atomic concepts and forming complex descriptions, such as complement, conjunction and disjunction. A set of constants, dubbed as *individuals* and denoted with $N_I = \{a, b, \cdots\}$, is to be considered as the names of the objects of the domain to be represented.

The semantics of the constructs is defined in terms of *interpretations*. An interpretation is a couple $\mathcal{I} = (\Delta, \cdot^{\mathcal{I}})$, where its *domain* Δ^I is a non-empty set of objects while $\cdot^{\mathcal{I}}$ is the *interpretation function* that maps each concept $C \in N_C$ onto a set of objects $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each role $R \in N_R$ onto a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. In addition, $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$ and $\bot^{\mathcal{I}} = \emptyset$. The semantics of complex concept descriptions is defined recursively depending on the available operators for building complex concepts. For instance, for the case of $\mathcal{ALC}$, the semantics of complex description is defined as follows:

- $(D \sqcap E)^{\mathcal{I}} = D^{\mathcal{I}} \cap E^{\mathcal{I}}$
- $(\neg D)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus D^{\mathcal{I}}$
- $(\forall R.D)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} | \forall b \in \Delta^{\mathcal{I}},\ (a, b) \in R^{\mathcal{I}} \to b \in D^{\mathcal{I}}\}$
- $(\exists R.D)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} | \exists b \in \Delta^{\mathcal{I}},\ (a, b) \in R^{\mathcal{I}} \land b \in D^{\mathcal{I}}\}$

Finally, each individual name is mapped onto an element of $\Delta^{\mathcal{I}}$.

A knowledge base is a pair $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ where $\mathcal{T}$ and $\mathcal{A}$ denote its TBox and ABox. The TBox contains intensional knowledge about the domain, modeled as inclusion axioms $C \sqsubseteq D$ (meaning that D subsumes C) and interpreted as $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for every interpretations $\mathcal{I}$. Given two concepts C and D, C is equivalent to D if for every interpretations $\mathcal{I}$, $C^{\mathcal{I}} = D^{\mathcal{I}}$. Alternatively, C and D are equivalent if $C \sqsubseteq D$ and $D \sqsubseteq C$. The ABox $\mathcal{A}$ contains factual knowledge,

[1] www.w3.org/OWL

i.e. assertions concerning individuals. In the ABox there are two kinds of assertions: *concept* $C(a)$ and *role assertions* $R(a, b)$. The set of individuals occurring in $\mathcal{A}$ are denoted by $\mathsf{Ind}(\mathcal{A})$.

Knowledge bases are also equipped with deductive reasoning capabilities. An important reasoning service for our purposes is *instance checking*: an individual a is an instance of a concept C if, for every model of $\mathcal{K}$, $C(a)$ holds. This can be denoted with $\mathcal{K} \models C(a)$. We will be also interested in the case where $\mathcal{K} \models \neg C(a)$. These instances will be exploited as *examples* (positive and negative examples respectively) in our learning procedures. Note that, due to the reasoning under the *Open World Assumption* (OWA) that is generally adopted in this context, it may happen that $C(a)$ and $\neg C(a)$ are satisfied by different models of $\mathcal{K}$. This means that neither $\mathcal{K} \models C(a)$ nor $\mathcal{K} \models \neg C(a)$ holds, i.e. there is insufficient knowledge to decide the membership of a w.r.t. the target concept using standard deductive inference services. Such individuals will be considered as instances with *uncertain membership* w.r.t. C.

In order to overcome this inherent limitation, it is possible to resort to decision procedures that are based on inductive (statistical) classification models. They can be learned by fitting a function from available examples (individuals for which the membership w.r.t. C is known) that amounts to solving a minimization problem based on a notion of misclassification *risk*. A general learning task aiming at classification models can be defined as follows:

Definition 1 (learning problem).
Given

- *a* target concept C
- *a set of instances* $\mathbf{E}$
- *a set of* labels $\mathcal{L}$ *to denote the membership w.r.t.* C
- *a joint* probability distribution *between* $\mathbf{E}$ *and* $\mathcal{L}$*, namely* $P(\mathbf{E}, \mathcal{L})$*, measuring the chance of an element of* $\mathbf{E}$ *to be assigned with one of the labels*

- *a set of* hypotheses $\mathcal{H} = \{h : \mathbf{E} \to \mathcal{L}\}$, *i.e.* classification functions *that can predict a label for their arguments*

- *a* loss function $L : \mathbf{E} \times \mathcal{L} \to [0, +\infty[$ *to assign a penalty for predicting an incorrect label for a given instance*

Find *a function $h^* \in \mathcal{H}$ such that:*

$$h^* = \arg\min_{h \in \mathcal{H}} \mathbb{E}_P \left[L(h(a), l) \right] \tag{1}$$

This definition requires the expected risk to be computed over the data generating distribution P, which is usually unknown. Therefore a more concrete definition will be specified for the case of DL knowledge bases, aimed at inducing a classification function that minimizes an *empirical risk* of error on the training set, and it can be reformulated for the targeted representation as follows:

Definition 2 (learning classifiers for DL knowledge bases).
Given

- *a* target concept C *in the signature of a knowledge base* $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

- *a set of* membership labels $\mathcal{L} = \{-1, 0, +1\}$ *to denote, resp., the positive, uncertain and negative membership w.r.t.* C

- *a* loss function $L : \mathsf{Ind}(\mathcal{A}) \times \mathcal{L} \to [0, +\infty[$

- *a* training set *of* examples *for which the correct labels are known, i.e. the values of a correct classifier*[2] $f : \mathsf{Ind}(\mathcal{A}) \to \mathcal{L}$, $\mathbf{Tr} = \mathbf{P} \cup \mathbf{N} \cup \mathbf{U}$ *where:*

  $\mathbf{P} = \{a \in \mathsf{Ind}(\mathcal{A}) \mid f(a) = +1\}$ *i.e.* $\{a \in \mathsf{Ind}(\mathcal{A}) \mid \mathcal{K} \models C(a)\}$

  $\mathbf{N} = \{a \in \mathsf{Ind}(\mathcal{A}) \mid f(a) = -1\}$ *i.e.* $\{a \in \mathsf{Ind}(\mathcal{A}) \mid \mathcal{K} \models \neg C(a)\}$

  $\mathbf{U} = \{a \in \mathsf{Ind}(\mathcal{A}) \mid f(a) = 0\}$ *i.e.* $\{a \in \mathsf{Ind}(\mathcal{A}) \mid \mathcal{K} \not\models C(a) \land \mathcal{K} \not\models \neg C(a)\}$

- *a set of classification functions, or* hypotheses, $\mathcal{H} = \{h : \mathsf{Ind}(\mathcal{A}) \to \mathcal{L}\}$

[2] i.e. whose analytic form is not available.

Find *a* classification function $h^* \in \mathcal{H}$*, approximating* f*, such that*

$$h^* = \arg\min_{h \in \mathcal{H}} \frac{1}{|\mathbf{Tr}|} \sum_{a \in \mathbf{Tr}} L\left[h^*(a), f(a)\right] \tag{2}$$

Note that the hypothesis set $\mathcal{H}$ acts as a form of *bias* and can be properly defined in order to exclude trivial solutions (overfitting) such as classifiers induced by a *rote learner* based on functions that merely memorize the correct labeling for the training examples (that would be equivalently described by the disjunction of very specific concepts – one per positive example). Conversely, the aim is to obtain a solution that is able to ensure a good generalization, that is the ability to correctly predict the membership for *unseen* individuals, i.e. individuals that have not been considered during the training phase. In this paper, we present a solution to this learning problem based on a tree classification model which combines logics and evidence-based prediction.

3. Basics of the Dempster-Shafer Theory

In this section basics of Dempster-Shafer Theory are summarized since it represents the main building block for the formalization of the evidential tree-based models presented in Sect. 4.

The Dempster-Shafer Theory (DST) [20] can be regarded as a generalization of the *Bayesian subjective probability* theory. The framework offers an alternative to traditional probabilistic theory for the mathematical representation of uncertainty: a probability mass can be assigned to a set or an interval without knowing the probability of the specific elements. As argued in [20], this aspect may be a valuable tool when knowledge is obtained from expert elicitation.

In the DST, the *frame of discernment* is a set of exhaustive and mutually exclusive hypotheses $\Omega = \{\omega_1, \omega_2, \ldots, \omega_n\}$ about a domain. Moving from the frame of discernment we can defined the *basic belief assignment*

Definition 3 (BBA and focal element). *A* basic belief assignment *(BBA) is defined as a mapping where* $m : 2^\Omega \to [0,1]$ *so that* $m(\emptyset) = 0$ *and* $\sum_{A \in 2^\Omega} m(A) = 1$*. If* $m(A) > 0$*,* A *is a* focal element *for* m*.*

Note that BBAs extend the standard probability measures. The key difference regards the relaxation of the *monotonicity* property of the probability measures. Indeed the degrees of belief are ascribed to sets of events rather than to single events. This means that for a BBA, given $A, B \in 2^{\Omega}$, $A \subseteq B$ does not imply $m(A) \leq m(B)$. This property derives from committing the value $m(B)$ only to the set B and not to any of its subsets. Conversely, in the probability theory, the probability of an event A is exactly the sum of probabilities assigned to the single $a \in A$.

Other functions can be derived from BBAs such as *belief* and *plausibility.*

Definition 4 (belief). *The* belief *in A, denoted by* Bel(A)*, represents a measure of the support committed to A given the available evidence:*

$$\mathrm{Bel}(A) = \sum_{B \subseteq A} m(B).$$

Definition 5 (plausibility). *The* plausibility *of A,* Pl(A)*, represents the total belief that may be committed to A when further evidence becomes available:*

$$\mathrm{Pl}(A) = \sum_{B \cap A \neq \emptyset} m(B).$$

Note that, differently from the BBA m, Bel and Pl are monotonic. As described in the following, this is taken into account when these measures are used with the models proposed in this paper.

3.1. Combination Rules

Combination rules are operators for pooling information obtained from multiple sources. These sources provide different assessments for the frame of discernment of the domain of interest. DST traditionally assumes that these sources are independent, although this constraint has been progressively relaxed with the introduction of new rules.

Many operations have been proposed in the literature [19]. In the sequel, we briefly survey the most important combination rules. In the rest of the paper, we will denote the application of one of such combination rules on two (or multiple) BBAs with the symbol $\oplus$ (e.g. $m_{1,2} = m_1 \oplus m_2$).

3.1.1. Dempster's Rule

The original combination rule of multiple BBAs known as Dempster's rule is a generalization of Bayes' rule [21]. The resulting BBA can be computed with:

$$\forall A \subseteq \Omega \qquad m_{1,2}(A) = \begin{cases} \frac{1}{1-c} \sum_{B \cap C = A} m_1(B) \cdot m_2(C) & \text{if } A \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where

$$c = \sum_{B \cap C = \emptyset} m_1(B) m_2(C) \tag{4}$$

This rule emphasizes the agreement between the sources adopting the normalizing factor c to distribute the conflicting evidence. It has come under serious criticism when the amount of conflict among sources is significant leading to counterintuitive results.

Example 1 (Dempster's rule). *Let us consider two BBAs m_1 and m_2 defined over a simple frame of discernment $\Omega = \{\omega_1, \omega_2, \omega_3\}$ whose focal elements are reported below:*

$$m_1(\{\omega_1\}) = 0.99, \; m_1(\{\omega_3\}) = 0.01, \; m_2(\{\omega_2\}) = 0.99, \; m_2(\{\omega_3\}) = 0.01$$

Applying the rule, the pooled BBA value for $\{\omega_3\}$ is:

$$m_{1,2}(\{\omega_3\}) = \frac{1}{1 - 0.99 \cdot 0.99 - 0.99 \cdot 0.01 - 0.99 \cdot 0.01} \cdot 0.01 \cdot 0.01 = 1$$

Note that this result is due to the agreement of the evidence in favor of $\{\omega_3\}$ and the disagreement between $\{\omega_1\}$ and $\{\omega_2\}$.

To prevent cases like the one reported above, which may affect the effectiveness of the models described in this paper, we investigated the effectiveness of further rules.

3.1.2. Dubois-Prade Disjunctive Pooling Rule

This rule [22] takes into account the union of the probability masses (disjunctive rule): this prevents the generation of conflict as there is no rejection of information coming from the various sources. The combination rule is defined as follows:

$$\forall A \subseteq \Omega \qquad m_{1,2}(A) = \sum_{B \cup C = A} m_1(B)m_2(C). \tag{5}$$

The union does not generate any conflict and does not reject any information asserted by the sources. As such, no normalization is required (unlike the Dempter's rule). The drawback of this rule is that it may yield a more imprecise result than desirable. It is easy to see that this rule is commutative and associative.

3.1.3. Mixing

This rule (also known as *averaging*) represents an extension of the average for probability distributions computed on the BBAs and describes the frequency of the various values within a range of possible values. The resulting BBA can be obtained merely as a weighted average of the masses according to the various features:

$$\forall A \subseteq \Omega \qquad m_{1,\ldots,n}(A) = \frac{1}{n} w_i m(A) \tag{6}$$

where a normalized weight vector $\mathbf{w}$ is generally considered. The values of the weights should reflect a degree of confidence in the sources. This rule is commutative, idempotent and quasi-associative[3]. For our purpose, we are interested in associative combination rules to prevent the final decision to be affected by the pooling order of the considered BBAs, namely *Dempster's rule* and *Dubois-Prade rule*. In the experiments we will consider also mixing rule to investigate

[3] A quasi-associative operation is an operation that can be broken down in two associative sub-operations. For instance, the mathematical average is quasi-associative: the value is obtained as the sum of a list of numbers divided by the number of the elements in the list (both the sum of the terms and the counting of the element in the list are associative operations)

the effectiveness of the predictive models when a quasi-associative rule is employed.

3.2. Measures of Total Uncertainty

In the context of the DST, various measures of uncertainty can be considered. These measures are typically defined as generalizations of *Shannon's entropy* or of other types of measures of uncertainty proposed in Probability Theory. Alternatively, they can be determined according to the conflict existing among the BBAs to be pooled according to a given combination rule. In this section, we briefly recall some measures. For more details, see [8].

The *non-specificity* measure [23] quantifies the degree of imprecision related to a BBA:

$$\mathrm{NS}(m) = \sum_{A \in 2^{\Omega}} m(A) \log(|A|) \tag{7}$$

The measure of *confusion* is defined on the ground of a BBA and the belief measure, as reported below [24]:

$$\mathrm{Confusion}(m) = - \sum_{A \in 2^{\Omega}} m(A) \log(Bel(A)) \tag{8}$$

The measure of *dissonance* [25] is based on a BBA and the plausibility and is defined as follows:

$$\mathrm{Dissonance}(m) = - \sum_{A \in 2^{\Omega}} m(A) \log(Pl(A)) \tag{9}$$

In the sequel, we will adopt these criteria to select the best features that compose the model proposed in this paper.

4. Evidence-based Terminological Trees and Forests

The original notions of terminological decision trees and random forests will be now recalled before introducing the new methods for the induction and usage of the evidence-based versions of these classification models.

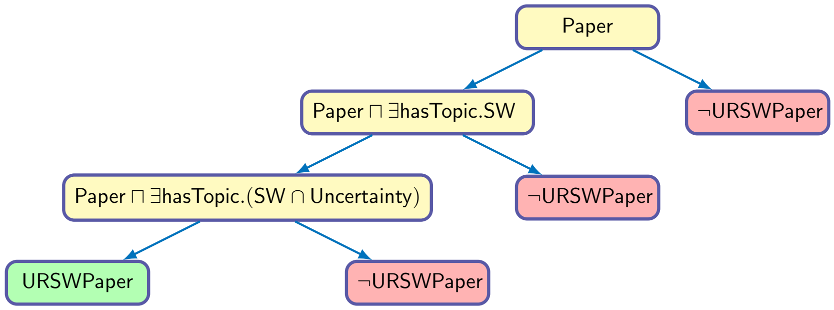

Figure 1: A TDT for predicting if a paper may have appeared in URSW proceedings

4.1. Terminological Decision Trees and Random Forests

Classification can be performed by inducing *terminological decision trees* (TDTs) [3]. A TDT is basically a binary tree whose leaves contain labels that denote the (positive/negative) membership with respect to the target concept; each inner node, dubbed also *decision* or *test* node, contains a DL concept description D (in conjunctive form) and the descending edges from such a node represent the result of a test over D (positive, negative).

Fig. 1 illustrates a simple example of a TDT that can describe the individuals of a knowledge base that are papers appeared in the URSW proceedings (target concept $\mathsf{URSWPaper}$). Note that, given a node with a concept description D, its left child may be either a leaf or another decision node containing a concept description E such that $E \sqsubseteq D$, whereas the right child may be either another leaf or a decision node containing a concept description E' for which $E' \sqsubseteq \neg D$ is intended. For instance, the root node contains the concept Paper while its left child is another decision node containing the concept description $\mathsf{Paper} \sqcap \exists \mathsf{hasTopic.SW}$ and its right child is a leaf with a negative label.

Similarly to other supervised models, the predictiveness of a TDT can be affected by the *class-imbalance problem*. In Machine Learning, this problem concerns the skewness of the training data distributions. Especially in multi-label settings, the problem occurs when the number of training examples belonging to a particular category (the *majority class*) overwhelms the number of those belonging to the others.

In order to tackle this problem, the most common approaches that have been proposed are based on a sampling strategy [26]. One of the simplest methods is an *under-sampling* strategy that randomly discards instances belonging to the majority class in order to re-balance the dataset. However, this method causes a loss of information due to the possible removal of useful (*critical*) examples that may be essential for inducing a predictive model.

A *terminological random forest* (TRF) is an ensemble model trained through a procedure that combines a random under-sampling strategy with ensemble learning [10]. A TRF is basically made up of a certain number of TDTs, where each of them is built by considering a (quasi-)balanced dataset. The ensemble model assigns the final classification for a new individual by appealing to a majority vote procedure. Therefore each TDT returns a *crisp prediction*: each provides an equal contribution to the final decision regarding the membership label, as no measure of confidence is available per single prediction.

In order to consider also this kind of information and tackling other relevant problems related to the uncertainty about the class assignments (e.g. cases of ties in conflicting predictions) and the disagreement between classifiers that may lead to misclassifications [10], we need to resort to other models for the ensemble approach.

4.2. Evidential Terminological Decision Trees

To better take into account the mentioned forms of uncertainty, it has been shown how approximate class-membership prediction can be carried out by inducing *evidential terminological decision trees* (ETDTs) [7], an extension of the TDTs based on the DST. ETDTs are defined in a similar way with respect to TDTs. However, unlike TDTs, each inner node contains a pair $\langle D, m \rangle$ where, besides the concept description D, there is a BBA m based on the membership of the individuals w.r.t. D.

Fig. 2 reports an example of ETDT used for deciding whether a paper has been published in the proceedings of URSW. Similarly to a TDT, each decision node contains a concept description D, while the left (resp. right) child may

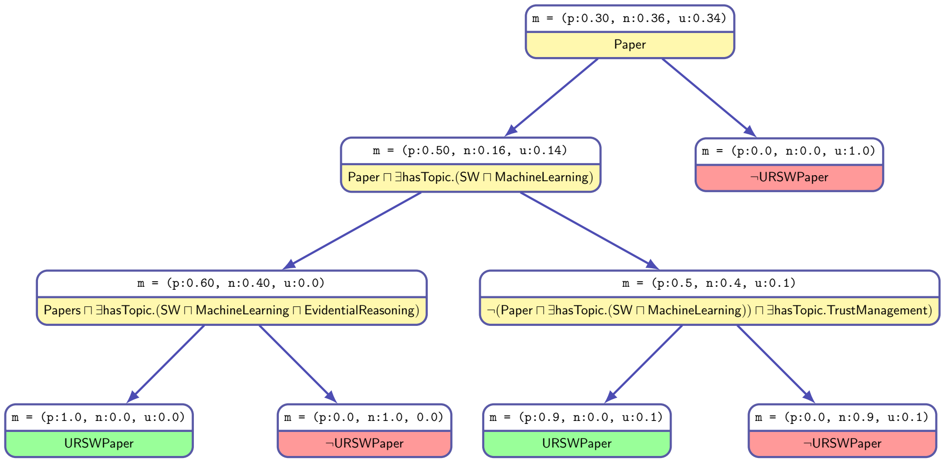

Figure 2: An ETDT for deciding if a paper has appeared in the URSW proceedings

either be a leaf (containing the corresponding label) or another decision node with a concept description $E \sqsubseteq D$ (resp. $E \sqsubseteq \neg D$). In addition, each node contains also a BBA, which can be estimated from the training instances used to learn the model, as described in the sequel.

4.2.1. Growing ETDTs

Before presenting the learning procedure we need to introduce the some notation. Moving from the formulation of the learning problem [10] (defined in Sect. 2), we will use the subset of the definite classification labels, $\Omega = \{-1, +1\} \subset \mathcal{L}$, as the *frame of discernment* of the problem (see Sect. 3). Therefore the positive membership label $+1$ corresponds to the subset $\{+1\}$ of the frame of discernment, the negative membership label -1 corresponds to the subset {-1}, and the case of uncertain-membership will be denoted with the label 0 corresponding to $\{-1, +1\}$.

Practically, to learn an ETDT model, a *divide-and-conquer* approach is adopted where a set of (more specific) concept descriptions is generated from the one contained in parent nodes. For each specialization, a BBA is also computed. Then the best description (and the corresponding BBA) is selected, e.g.

Algorithm 1 The routines for inducing ETDTs

```
const θ ∈ ]0, 1] {min.\ purity threshold parameter}

function InduceETDTree(⟨P, N, U⟩, C, D, m, P̂r): T
input ⟨P, N, U⟩: training set; C: target concept; D: concept,m: BBA; P̂r: priors
output T: ETDT
begin
T ← new ETDT
if |P| = 0 and |N| = 0 then
    if P̂r(+1) ≥ P̂r(−1) then {pre−defined constants wrt the whole training set}
        T.root ← ⟨C, m⟩
    else
        T.root ← ⟨¬C, m⟩
else if (m({−1} ≃ 0) and (m({+1}) > θ) then
    T.root ← ⟨C, m⟩
else if (m({+1} ≃ 0) and (m({−1}) > θ) then
    T.root ← ⟨¬C, m⟩
else
    S ← ∅
    for E ∈ ρ(D) {assignBBA for each candidate}
        m_E ← computeBBA(E, ⟨P, N, U⟩)
        S ← S ∪ {⟨E, m_E⟩}
    ⟨E*, m*⟩ ← selectBestCandidate(S)
    ⟨⟨P^l, N^l, U^l⟩, ⟨P^r, N^r, U^r⟩⟩ ← split(E*, ⟨P, N, U⟩)
    T.root ← ⟨E*, m*⟩
    T.left ← induceETDT(⟨P^l, N^l, U^l⟩, C, E*, m*, P̂r)
    T.right ← induceETDT(⟨P^r, N^r, U^r⟩, C, ¬E*, m*, P̂r)
return T
end
```

the one having the smallest non-specificity w.r.t. the previous level.

Alg. 1 illustrates the training procedure. It distinguishes various cases: the non-recursive ones are those for which leaves are defined while the final one determines the inner nodes, hence the subtree structure, recursively.

The first case copes with the lack of examples ($|\mathbf{P}| = 0$ and $|\mathbf{N}| = 0$) routed to the node resorting to the prior probability (estimates).

The following cases determine the label for a leaf-node when it is (sufficiently) pure, i.e. no positive (resp. negative) example is found (or just a few) while most of the examples are negative (resp. positive). This *purity* condition is evaluated by considering the BBA m given as an input to the algorithm ($m(\{-1\} \simeq 0$ and $m(\{+1\}) > \theta$ or $m(\{+1\} \simeq 0$ and $m(\{-1\}) > \theta$), where θ is a purity threshold. The values of a BBA function for the membership values are obtained from the distribution of positive, negative and uncertain-membership instances w.r.t. the current concept.

Finally, the last (recursive) case concerns the availability of a nonnegligible number of both negative and positive examples. In this case, the current concept description D has to be specialized by means of an operator exploring the search space of downward refinements of D. Following the approach described in [10, 12], the refinement step produces a set of candidate specializations $\rho(D)$. A BBA m_E is then built for each candidate $E \in \rho(D)$. Again, the function can be obtained by counting the number of positive, negative and uncertain-membership instances). Then the best pair $\langle E^*, m^* \rangle \in \mathbf{S}$ according to the non-specificity measure is determined by the SELECTBESTCANDIDATE procedure and finally installed in the current node. Specifically, the procedure tries to find the pair $\langle E^*, m^* \rangle$ having the smallest non-specificity measure. As an alternative, the best concept description can be selected in order to maximize either confusion measure or the dissonance measure w.r.t. the previous level.

After the assessment of the best test concept description E^*, the individuals are partitioned by the procedure SPLIT for the left or right branch according to the result of the test w.r.t. E^*, maintaining the same group[4] ($\mathbf{P}^{l/r}$,$\mathbf{N}^{l/r}$, or $\mathbf{U}^{l/r}$). Note that a training example a is replicated in both children in case both $\mathcal{K} \not\models E^*(a)$ and $\mathcal{K} \not\models \neg E^*(a)$ (test with a non-definite, positive or negative, outcome). The divide-and-conquer strategy is applied recursively until the instances routed to a node satisfy one of the stopping conditions discussed above.

From a *learning-as-search* perspective, one may regard the induction of an ETDT as a search process in a hypothesis space $\mathcal{H}$ defined by the set of all possible ETDTs ruling out those having a sole inner node in the form of a pair $(\top, m)$.

4.2.2. Prediction

Given a test individual a and the induced ETDT, the membership can be assessed by following one or more paths in the tree. The procedure is reported

[4] Note that the group is related to the membership w.r.t. the target class, while the branch direction depends on the outcome of the test w.r.t. E^*.

Algorithm 2 Class-membership prediction routine through ETDT

const $\varepsilon \in]0, 1]$ {decision threshold parameter}

function $\textsc{classifyByETDT}(a, T) : l$
input a: individual; T: ETDT
output $l \in \mathcal{L}$
begin
$M \leftarrow \textsc{getLeafBBAList}(a, T)$ {list of BBAs located at leaf−nodes}
$\bar{m} \leftarrow \bigoplus_{m \in M} m$
for each $\emptyset \neq s \in 2^{\Omega}$ **do**
  Compute $\overline{Bel}(s)$ from $\bar{m}$
if $|\overline{Bel}(\{-1\}) - \overline{Bel}(\{+1\})| \leq \varepsilon$ **then**
  predlabel $\leftarrow 0$ {case of uncertain membership}
else
  predlabel $\leftarrow \arg\max_{l \in \Omega} \overline{Bel}(\{l\})$ {cases of definite membership}
return predlabel
end

in Alg. 2.

Specifically, the algorithm traverses recursively the ETDT by performing a test w.r.t. the concept contained in each node that is reached: let $a \in \mathsf{Ind}(\mathcal{A})$ and D the concept installed in the current node, if $\mathcal{K} \models D(a)$ (resp. $\mathcal{K} \models \neg D(a)$) the left (resp. right) branch is followed. If neither $\mathcal{K} \not\models D(a)$ nor $\mathcal{K} \not\models \neg D(a)$ is verified, both branches are followed.

After the exploration of an ETDT (via $\textsc{getLeafBBAList}$), the list M likely contains multiple BBAs. In this case, the BBAs are pooled according to a combination rule (see Sect. 3) producing $\bar{m}$.

The final decision about the membership to be assigned to the test individual is made by computing the belief measures for the positive, negative and uncertain membership cases based on the pooled BBA. If the measures for the definite cases are approximately equal (their difference is below a given threshold ε), the algorithm will assign the uncertain membership label 0. Conversely, the algorithm selects the definite label ($l \in \Omega$) with higher belief.

4.3. Evidential Terminological Random Forests

An *evidential terminological random forest* (ETRF) is an ensemble of ETDTs. We will focus on the procedures for producing an ETRF and for predicting class-membership of input individuals exploiting an ETRF.

Algorithm 3 The routines for inducing ETRFs

const $\theta \in]0, 1]$ {min.\ purity threshold parameter}

function INDUCEETRF($\mathbf{Tr}, C, n$): $\mathbf{F}$
input $\mathbf{Tr}$: training set; C : target concept; $n \in \mathbb{N}$
output $\mathbf{F}$: ETRF
begin
$\hat{\Pr} \leftarrow$ ESTIMATEPRIORS($\mathbf{Tr}, C$): {C prior membership probability estimates}
$\mathbf{F} \leftarrow \emptyset$
parfor $i \leftarrow 1$ **to** n
    $\mathbf{D}_i \leftarrow$ BALANCEDBOOTSTRAPSAMPLE($\mathbf{Tr}$)
    let $\mathbf{D}_i = \langle \mathbf{P}, \mathbf{N}, \mathbf{U} \rangle$
    $m_i \leftarrow$ COMPUTEBBA($C, \langle \mathbf{P}, \mathbf{N}, \mathbf{U} \rangle$)
    $T_i \leftarrow$ INDUCEETDTREE($\mathbf{D}_i, C, \top, m, \hat{\Pr}$);
    $\mathbf{F} \leftarrow \mathbf{F} \cup \{T_i\}$
return F
end

4.3.1. Growing ETRFs

Alg. 3 describes the procedure for producing an ETRF. To this purpose, the target concept C, a training set $\mathbf{Tr} \subseteq \mathsf{Ind}(\mathcal{A})$ and the desired number of trees n are required. $\mathbf{Tr}$ may contain not only positive and negative examples but also instances with uncertain membership w.r.t. C.

Similarly to a bagging approach, the training individuals are sampled with replacement in order to obtain n subsets $\mathbf{D}_i \subseteq \mathbf{Tr}$, with $i = 1, \ldots, n$. It is possible to apply various sampling strategies to obtain the various samples $\mathbf{D}_i$. In this study we followed the same approach already used in our previous work [10]. Firstly, the initial data distribution is considered by adopting a stratified sampling strategy w.r.t. the class-membership values to ensure the availability of instances of the minority class. In the second phase, undersampling can be performed on the training set in order to obtain (quasi-)balanced $\mathbf{D}_i$ sets (i.e. with a class imbalance that will not affect much the training process). This means that if the majority class is the negative one, the exceeding part of the negative examples is randomly discarded. In the dual case, positive instances are removed. In addition, the sampling procedure removes also all the instances of uncertain membership.

In Alg. 3, procedure BALANCEDBOOTSTRAPSAMPLE implements this strategy returning the samples $\mathbf{D}_i$. For each $\mathbf{D}_i$, an ETDT T is built by invoking the procedure INDUCEETDT. Note that, the procedure for learning an ETDT

Algorithm 4 Class-membership prediction routines for ETRFs

```
const ε ∈ ]0,1] {decision threshold parameter}

function classifyByETRF(a, F, C) : l
input a: individual; F: ETRF; C: target concept
output l ∈ ℒ
begin
M[] ← new map {trees to BBAs}
parfor each T ∈ F do
    M[T] ← getTreeBBA(a, T)
m̄ ← ⊕_{m∈M} m {pooling according to a combination rule}
for each ∅ ≠ s ∈ 2^Ω do
    Compute Bel̄(s) from m̄
if |Bel̄({−1}) − Bel̄({+1})| ≤ ε then
    predlabel ← 0 {case of uncertain membership}
else
    predlabel ← arg max_{l∈Ω} Bel̄({l}) {cases of definite membership}
return predlabel
end

function getTreeBBA(a, T) : m̄
input a: individual; T: ETDT
output m̄: BBA
begin
M ← getLeafBBAList(a, T) {list of BBAs}
m̄ ← ⊕_{m∈M} m
return m̄
end
```

for the forest requires the introduction of some further amount of *randomization*: the recursive case of Alg. 1 was modified so that the computation of the BBAs and the selection of the best refinement are made considering a subset $\mathbf{RS} \subseteq \rho(D)$ of randomly selected candidate specializations. This may be crucial to improve the performance w.r.t. the one of a single classifier through a good diversification among the trees.

4.3.2. Prediction

Given an ETRF, predictions can be made relying on the resulting classification model. The related procedure, sketched in Alg. 4, works as follows.

Given the individual to be classified, for each tree T_i of the forest, the procedure GETTREEBBA returns a BBA obtained by pooling the various BBAs found at the leaves reached from the root in a traversal path down the tree.

After polling all the trees in the ensemble, a set of BBAs deriving from the previous phase are exploited to decide the classification for the test individual a.

Function classifyByETRF takes an individual a and a forest F. Then, the algorithm iterates on the forest trees collecting the BBAs via function getTreeBBA.

Then, the BBAs are pooled according to a further combination rule, which can be different from the one employed during the exploration of a single ETDT. Additionally, this combination rule should be also an associative operator [19]. In this way, the result should not be affected by the pooling order of the BBAs.

In [18] we combined these BBAs via Dempster's rule. Using this rule, the disagreement among the classifiers, that corresponds to the conflict exploited as a normalization factor, is explicitly considered by the meta-learner. Again, the final decision is then made according to the belief function value computed from the pooled BBAs $\overline{m}$.

4.4. Simplifying the Ensemble

In the previous works [10, 18], we noticed that a limited number of ETDTs was usually sufficient to obtain a good performance. Growing forests with larger numbers of trees did not improve significantly on predictiveness (in some cases the performance even worsened). Moreover, the efficiency of the induction and classification procedures obviously decayed owing to the increased number of trees. Therefore, in this section, we illustrate how DST constructs can support the simplification of an ETRF to increase the efficiency of the classification phase while preserving its effectiveness.

The proposed solution (see Alg. 5) assumes that the prediction made using an ETRF of progressively increasing size may lead to a poorer (or similar) performance depending on the amount of conflictual evidence coming from a larger number of trees. This basically implies that the confidence in the predictions may decrease up to some point when the resulting predictions may even differ from the expected ones.

The algorithm for pruning the ensemble is incremental, this means that it works by considering one tree at a time. Specifically, given a forest $\mathbf{F}$, the algorithm produces a new forest $\mathbf{F}'$ as follows: it combines the pooled BBAs

Algorithm 5 Conflict-based ensemble simplification

function SIMPLIFICATION($\mathbf{F}$) : $\mathbf{F}'$
input $\mathbf{F}$: TRF
output $\mathbf{F}'$: TRF
begin
$M[] \leftarrow$ **new** array
for each $T \in \mathbf{F}$ **do**
    $M[T] \leftarrow$ GETBBAFROMTREE(T)

$\overline{m} \leftarrow M[T_1]$
$\mathbf{F}' \leftarrow \{\}$ {initialize with the first ETDT in the forest}
for each $T \in \mathbf{F}$ **do**
    $c \leftarrow \sum_{B \cap C = \emptyset} \overline{m}(B)(M[T])(C)$
    if $c \leq \nu$ **then**
        $\mathbf{F}' \leftarrow \mathbf{F}' \cup \{T\}$
        $\overline{m} \leftarrow \overline{m} \oplus M[T]$

if $\mathbf{F}' = \emptyset$ **then**
    $\mathbf{F}' \leftarrow \mathbf{F}' \cup \{T_1, T_2\}$ {return a forest with size=2, in case of oversimplification}

return $\mathbf{F}'$
end

coming from each ETDT in the forest in order to compute the conflict measure c (see Eq. 3). If the conflict does not go beyond a given threshold, namely ν, the current tree T is added to $\mathbf{F}'$.

The BBA drawn from a $T \in \mathbf{F}'$ and returned to the main procedure is computed as follows: T is traversed following all the possible paths until all the leaves are reached in order to collect the BBAs. Subsequently, the BBAs are combined according to an associative rule (to avoid order-dependent results). This is implemented in the procedure GETBBAFROMTREE. The resulting BBA is then returned to the main procedure and used to determine c.

A particular case occurs when the conflict exceeds the threshold ν. In this case, to prevent the production of an empty ETRF, the algorithm returns a default forest composed by only two ETDTs.

5. Empirical Evaluation

The evaluation reported in this section aimed at assessing the effectiveness of ETRFs and ETDTs proposed in this paper[5].

[5]The source code is available at: https://github.com/Giuseppe-Rizzo/SWMLAlgorithms

Table 1: Ontologies employed in the experiments

Ontology	DL Lang.	#*Axioms*	#*Concepts*	#*Roles*	#*Individuals*
BCO	$\mathcal{ALCHOF}(\mathcal{D})$	1098	196	22	112
BioPax	$\mathcal{ALCIF}(D)$	2617	74	70	323
NTN	$\mathcal{SHIF}(D)$	1516	47	27	676
HD	$\mathcal{ALCIF}(D)$	8811	1498	10	639
Financial	$\mathcal{ALCIF}(D)$	3509	60	16	1000
monetary	$\mathcal{ALCIF}(D)$	7562	323	247	2466
DBpedia	$\mathcal{ALCH}$	78663	251	132	16606

Table 2: Distribution of the positive, negative and uncertain instances w.r.t the artificially generated target concepts for the various ontologies considered in the experiments

Ontology	% Pos.	% Neg.	% Unc.
BCO	17	53	30
Biopax	40	40	20
NTN	24	13	63
HD	24	11	65
Financial	26	47	30
Monetary	36	44	20
DBPedia	16	14	70

5.1. Setup of the Experimental Sessions

The experiments have been carried out on various Web ontologies (see Tab. 1) that are available on public repositories[6]. For each ontology, 15 query concepts have been randomly generated by combining 2 through 8 (primitive or defined) concepts of the ontology (using the conjunction and disjunction operators or universal and existential restrictions). Each concept was generated so that at least 40 positive examples and 40 negative examples can be found among the individuals of the knowledge base.

Tab. 2 illustrates the average rate of the positive, negative, uncertain examples (computed considering all the individuals of $\mathsf{Ind}(\mathcal{A})$) over the number of query concepts.

[6]See http://owl.cs.manchester.ac.uk/tools/repositories/

We compared the methods and models proposed in this paper with a variety of other approaches in the literature related to the task of inductive classification with DL knowledge bases. Specifically, we selected:

- purely logical approaches, such as TDTs [3], CELOE [27], TRFs [10] and the previous versions of ETDTs [7] and ETRFs;
- an instance-based method, i.e. the k-nearest neighbor algorithm embedding a suitable distance measure as illustrated in [1];
- a kernel method for linear models, i.e. the *kernel perceptron* [28] adopting a kernel function for individuals in DL knowledge bases [29, 1].

In the experiments with the ETDTs the three total uncertainty measures reported in Sect. 3.2 have been considered: *non-specificity*, *confusion* and *dissonance*. We repeated the experiments varying also the combination rules for pooling the BBAs collected after tests with uncertain results are performed. The rules adopted in the evaluation were: Dempster's rule, *Dubois-Prade's* rule and the *mixing* rule.

The experiments on TRFs and ETRFs required a setup of the stratified sampling rate and the forest size. Three sampling rates have been picked, 50%, 70% and 80%, while the forest size has been set to 10, 20 and 30 trees. In the induction of (E)TDTs, the number of randomly selected specializations was determined as the square root of candidate refinements: $n(C) = \sqrt{\mid \rho(C) \mid}$. We ran the ETRF learning algorithm by varying three further parameters: the heuristics for inducing the ETDTs, the combination rule for pooling the BBA collected during the traversing process and the combination rules adopted as meta-learner. Besides, we performed experiments with the ETRFs induction algorithm with and without the simplification strategy, setting the threshold ν to 0.4. Also, we set the value of parameter ε (Alg. 2 and Alg. 4) to 0.3 for forcing the answer in favor of a definite membership, and the value for parameter θ (Alg. 1), used to control the growth of a tree (either a TDT or an ETDT), was heuristically[7] set to 0.9.

Concerning the k-NN algorithm, we set the neighborhood size to $k = \log |\mathsf{Tr}|$. The distance measure between individuals have been chosen from the family of measures proposed in [1] by setting its parameter p to 2 and using atomic concepts in the signature of the knowledge base as a feature set.

In the experiments with CELOE, we set a *noise rate* of 25% (representing the maximum number of admissible false negative cases).

Finally, the kernel perceptron required the choice of the *kernel function*, of the *learning rate* and of the *number of epochs* for the training phase. In the experiments, we used the kernel function between individuals of a DL knowledge base proposed in [29, 1], and we set a learning rate of 0.05 and a number of epochs of 200.

For each learning problem (each target concept considered for each dataset/ontology), we estimated the average performance of the models under comparison through a 10-fold cross validation procedure. The baseline (correct classification labels) for the various instances in the training and test sets w.r.t. the target concepts was computed by a DL reasoner. Specifically, the *macro-averaged* F_1-measure has been computed over the three membership values. In addition, the following indices have been measured [3, 10, 7].

- match rate (M%), the percentage of test individuals for which the inductive model agrees with the baseline (both positive, negative, or unknown);
- commission rate (C%), the fraction of test cases where the predicted membership is opposite w.r.t. the baseline (i.e. positive vs. negative or vice-versa);
- omission rate (O%), the proportion of test cases for which the inductive method cannot determine the definite membership that holds in the baseline (i.e. unknown vs. positive or negative);

[7] For each learning algorithm considered in the evaluation, the values have been tuned using a *leave-one-out* procedure.

- induction rate (I%). the percentage of test cases where the inductive method can predict a definite membership while it could not be determined for the baseline (i.e. positive or negative vs. unknown).

5.2. Outcomes

Table 3: Outcomes for ETDTs adopting the *mixing rule* in the classification step. The outcomes do not change significantly employing other combination rules

Ontology		Non-specificity	Dissonance	Confusion
BCO	F_1	83.56 ± 05.06	84.15 ± 06.14	84.15 ± 06.14
	M%	85.48 ± 11.01	91.31 ± 14.79	91.31 ± 14.79
	C%	07.56 ± 08.08	00.86 ± 02.61	00.86 ± 02.61
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	06.96 ± 05.97	07.83 ± 15.35	07.83 ± 15.35
BioPax	F_1	82.16 ± 08.32	82.43 ± 06.47	86.98 ± 08.32
	M%	86.63 ± 14.60	87.00 ± 07.15	87.00 ± 07.15
	C%	11.02 ± 12.95	11.57 ± 02.62	11.57 ± 02.62
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	02.35 ± 05.23	01.43 ± 08.32	01.43 ± 08.32
NTN	F_1	23.06 ± 26.14	14.65 ± 05.43	12.87 ± 26.54
	M%	23.87 ± 26.18	14.87 ± 24.18	13.85 ± 26.18
	C%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	75.13 ± 26.18	85.13 ± 24.18	86.15 ± 26.17
HD	F_1	85.48 ± 11.01	91.31 ± 14.79	91.31 ± 14.79
	M%	10.69 ± 01.47	10.69 ± 01.47	10.69 ± 01.47
	C%	00.07 ± 00.17	00.07 ± 00.17	00.07 ± 00.17
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	89.24 ± 01.46	89.24 ± 01.46	89.24 ± 01.46

Tables 3 through 17 present the outcomes of the various experiments. Preliminarily, note that for brevity, in the case of (E)TRFs, we report only the outcomes for ensemble models composed by 20 trees and induced using a 50% sampling rate as the performance had no significant variation in the experiments with the other values of such parameters. A similar consideration applies also to the experiments with (E)TDTs.

The results seem to be promising: ETDTs and ETRFs were competitive against other learning systems (see Tab. 3–4, 6–7, 8–9, 15–16). In some cases,

Table 4: Outcomes for ETDTs adopting the *mixing rule* in the classification step.

Ontology		Non-specificity	Dissonance	Confusion
	F_1	87.42 ± 08.23	88.23 ± 08.43	88.23 ± 08.43
	M%	83.43 ± 04.43	87.43 ± 17.42	87.43 ± 07.42
Financial	C%	04.00 ± 03.35	00.00 ± 00.00	00.00 ± 00.00
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	12.57 ± 13.45	07.53 ± 12.24	07.54 ± 12.24
	F_1	85.48 ± 11.01	91.31 ± 14.79	91.31 ± 14.79
	M%	87.43 ± 13.45	93.47 ± 12.24	93.46 ± 12.24
monetary	C%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	12.57 ± 13.45	07.53 ± 12.24	07.54 ± 12.24
	$F_!$	60.78 ± 23.08	60.78 ± 23.08	60.78 ± 23.08
	M%	53.84 ± 23.16	53.84 ± 23.16	54.46 ± 23.16
DBpedia	C%	35.28 ± 23.30	35.28 ± 23.30	35.28 ± 23.30
	O%	00.00 ± 00.00	00.00 ± 00.00	00.54 ± 00.03
	I%	10.86 ± 01.69	10.86 ± 01.69	10.72 ± 13.30

the new models outperformed the others in terms of match rate, especially k-NN and Perceptron. In general, we noted that the match rate obtained with ETDTs was particularly high for ontologies endowed with a large number of disjointness axioms, such as Biopax ,Monetary and, to some extent, Financial. The maximal average match rate were over 80% (for ETDTs) and over 90% (for ETRFs). As regards the F_1, it was particularly large on the aforementioned ontologies and improved in the experiments with ETRFs.

Table 5: Average run-time (secs) for the classification using ETDTs varying the combination rule

Ontology	Dempster	Dubois-Prade	Mixing
BCO	3.5	3.5	2.4
BioPax	7.5	7.5	7.5
NTN	4.5	4.5	2.5
HD	3.5	3.6	3
Financial	5	5	4.5
monetary	10.3	10.3	10.3
DBpedia	10	10	4

5.2.1. ETDTs

In the experiments with the ETDTs (see Tab. 3), the match rate was larger and the commission rate was smaller using either the confusion or the dissonance measure with respect to the outcomes observed when the non-specificity measure was adopted. This was likely due to the fact that, adopting the non-specificity measure, the heuristic basically tended to select concepts with a definite membership w.r.t. the target concept, with little or no increase of homogeneity in the child nodes. As a consequence, even extending the branches with more descendants, no significant gain was observed and the resulting ETDTs tended to overfit the training instances. Conversely, adopting the confusion and dissonance, the algorithm was biased towards shorter (and more predictive) trees where pure leaves were obtained more easily.

As regards the employment of different combination rules to classify individuals through ETDTs, we observed that none led to significant improvements. On one hand, in the case of ontologies with properly defined constraints such as concept disjointness, e.g. BioPax, the classification procedure tended to traverse single branches thanks to intermediate tests with definite decisions. This low degree of uncertainty yielded an analogous behavior w.r.t. the case of TDTs and, consequently, to similar outcomes. On the other hand, in the case of ontologies with a limited number of disjointness axioms more individuals exhibited an uncertain membership w.r.t. the test concepts, so the classification algorithm tends to traverse more branches reaching a larger number of BBAs (at the leaves): the pooled BBAs obtained through the three combination rules were very similar. Consequently, also the measures of belief used to decide the final classification did not change significantly. This suggested that quasi-associative rules, such as *mixing*, could be taken into account as alternative strategies for combining evidence (despite their being order-dependent) that are able to preserve the predictiveness of the classification models. This is an advantage because classification through ETDTs via *mixing rule* was more efficient than with the adoption of the other rules. This benefit was particularly

Table 6: Outcomes for the ETRFs obtained adopting the three heuristics for the best concept selection and Dempster's rule as meta-learner, with and without the use of the pruning

Ontology		No simplification			simplification		
		Non-specific.	Dissonance	Confusion	Non-specific.	Dissonance	Confusion
BCO	F_1	90.76 ± 06.67	91.76 ± 06.87	91.87 ± 07.23	95.23 ± 02.27	95.23 ± 02.27	95.23 ± 02.27
	M%	87.43 ± 09.13	88.23 ± 08.56	88.42 ± 08.43	92.31 ± 04.27	92.32 ± 04.27	92.31 ± 04.27
	C%	03.16 ± 03.09	02.44 ± 03.39	02.27 ± 03.38	02.81 ± 02.45	02.81 ± 02.45	02.91 ± 02.45
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	09.41 ± 03.56	09.33 ± 03.46	09.31 ± 03.43	04.88 ± 03.45	04.88 ± 03.45	04.88 ± 03.45
BioPax	F_1	90.37 ± 05.56	91.38 ± 05.57	92.43 ± 05.89	92.78 ± 05.84	92.78 ± 05.86	93.45 ± 04.57
	M%	93.45 ± 07.15	94.45 ± 07.14	94.45 ± 07.15	96.57 ± 06.15	95.98 ± 06.14	96.87 ± 06.23
	C%	05.22 ± 07.42	04.22 ± 07.42	04.22 ± 07.24	01.07 ± 01.67	01.71 ± 02.50	00.77 ± 01.74
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	01.33 ± 07.16	01.33 ± 07.16	01.97 ± 07.16	02.36 ± 04.24	02.31 ± 04.13	02.30 ± 08.15
NTN	F_1	04.15 ± 03.25	04.15 ± 03.25	04.15 ± 03.25	35.25 ± 03.87	35.23 ± 03.87	35.23 ± 03.87
	M%	05.50 ± 07.28	05.50 ± 07.28	05.50 ± 07.28	26.40 ± 05.15	26.43 ± 05.15	26.43 ± 05.14
	C%	06.52 ± 07.54	06.52 ± 07.54	06.52 ± 07.54	06.52 ± 07.54	06.52 ± 07.54	06.52 ± 07.54
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	87.99 ± 08.84	87.99 ± 08.84	87.99 ± 08.84	67.05 ± 05.35	67.05 ± 05.35	67.07 ± 05.35
HD	F_1	08.32 ± 00.15	08.32 ± 00.15	08.32 ± 00.15	28.15 ± 02.17	28.15 ± 02.17	28.15 ± 02.17
	M%	10.29 ± 00.00	10.29 ± 00.01	10.29 ± 00.02	32.56 ± 00.43	33.43 ± 00.43	33.56 ± 00.42
	C%	00.57 ± 00.05	00.57 ± 00.05	00.57 ± 00.05	00.14 ± 00.26	00.14 ± 00.27	00.14 ± 00.28
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	89.14 ± 00.26	89.14 ± 00.26	89.14 ± 00.26	67.44 ± 00.26	67.44 ± 00.26	67.44 ± 00.26

evident in the experiments with larger ontologies like DBpedia. For this ontology, on average the classification of an individual adopting the mixing rule was 60% faster than with the Dubois-Prade rule. Tab. 5 summarizes the average time (in seconds) required by an ETDT for classifying an individual.

5.2.2. ETRFs and simplification procedure

Concerning the experiments with the ETRFs (Tab. 6 - 8), as expected, the ensemble models showed on average a superior performance with respect to the ETDTs, and in most of the cases we observed a decrease of standard deviation. As previously mentioned, the F_1 increased with respect to the experiments with ETDTs, suggesting that the sampling strategy brought benefits to the predictiveness of ETRFs mitigating the of bias classification models towards the most

Table 7: Outcomes for the ETRFs obtained adopting the three heuristics for the best concept selection and Dempster's rule as meta-learner, with and without the use of the pruning

Ontology		No simplification			simplification		
		Non-specific.	Dissonance	Confusion	Non-specific.	Dissonance	Confusion
	F_1	90.14 ± 06.76	92.46± 07.43	96.79 ± 03.17	96.79 ± 03.17	96.79 ± 03.17	96.79 ± 03.17
	M%	93.43 ± 05.06	93.89 ± 05.16	94.03 ± 05.23	97.12 ± 03.10	97.13 ± 04.12	97.12 ± 04.15
Financial	C%	01.07 ± 01.67	01.71 ± 02.50	00.77 ± 01.74	00.60 ± 00.03	00.54 ± 00.03	00.77 ± 01.74
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	02.28 ± 08.13	02.31 ± 08.17	02.30 ± 08.15	02.28 ± 08.13	02.31 ± 08.17	02.30 ± 08.15
	F_1	95.23 ± 03.24	96.45 ± 03.76	96.57 ± 03.17	97.43 ± 02.14	97.43 ± 02.14	97.43 ± 02.14
	M%	93.43 ± 05.06	95.89 ± 05.16	94.56 ± 04.46	96.65 ± 04.35	99.43 ± 08.13	99.55 ± 08.15
monetary	C%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	06.57 ± 05.06	04.11 ± 05.16	01.97 ± 07.16	03.35 ± 04.35	00.57 ± 08.13	00.45 ± 08.15
	F_1	60.43 ± 02.15	60.43 ± 02.15	61.25 ± 02.24	68.34 ± 02.15	68.43 ± 02.15	68.34 ± 02.15
	M%	53.84 ± 05.43	53.84 ± 05.43	54.46 ± 05.43	70.44 ± 03.31	70.43 ± 03.31	70.44 ± 03.31
DBpedia	C%	00.08 ± 00.01	00.08 ± 00.02	00.08 ± 00.01	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	45.28 ± 23.30	45.28 ± 23.30	45.28 ± 23.30	29.56 ± 03.31	29.57 ± 03.31	29.56 ± 03.31

probable membership value. Such a stability of the ensemble models was likely due to the mediation operated by the the meta-learner over the various models, that positively influenced the final decision towards correct label assignments. Again, the choice of the combination rule for the BBAs at the leaves of a single ETDT did not affect significantly the performance of the considered tree models. Conversely, using further combination rules as meta-learners had a stronger influence on the performance. In particular, adopting Dubois-Prade rule we observed a decrease of the induction rate and an increase of the match rate. A similar outcome was obtained using the mixing rule. Unlike Dempster's rule, the adoption of the Dubois-Prade and mixing rules tended to reduce the evidence in favor of a definite membership. This means that the belief related to the hypotheses of positive and negative memberships were generally low and their difference often did not exceed the threshold ε.

A similar effect was observed in the experiments with the simplification method proposed in the paper: smaller ensembles tended to predict an

Table 8: Outcomes for ETRFs adopting the three heuristics for the best concept selection and the Dubois-Prade rule as a meta-learner, with and without the use of the pruning

Ontology		No simplification			simplification		
		Non-specific.	Dissonance	Confusion	Non-specific.	Dissonance	Confusion
BCO	F_1	87.13 ± 05.67	90.45 ± 03.56	90.48 ± 03.78	89.94 ± 02.13	89.13 ± 02.19	89.13 ± 02.15
	M%	90.44 ± 09.13	93.24 ± 08.56	93.40 ± 08.35	93.17 ± 04.27	94.45 ± 04.27	94.44 ± 04.15
	C%	03.16 ± 03.09	02.43 ± 03.39	02.29 ± 03.45	02.81 ± 02.45	02.81 ± 02.45	02.91 ± 02.45
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	06.40 ± 03.56	04.33 ± 03.27	04.31 ± 03.46	04.01 ± 03.45	02.73 ± 03.45	02.74 ± 03.56
BioPax	F_1	90.98± 03.79	92.45 ± 03.79	92.45 ± 03.79	93.76 ± 04.25	94.33 ± 05.25	94.33 ± 05.25
	M%	93.45 ± 07.15	94.45 ± 07.14	94.45 ± 07.15	96.57 ± 06.15	95.98 ± 06.14	96.87 ± 06.23
	C%	05.22± 07.42	04.22± 07.42	04.22± 07.24	01.07 ± 01.67	01.71 ± 02.50	00.77 ± 01.74
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	01.33 ± 07.16	01.33 ± 07.16	01.97 ± 07.16	02.36 ± 04.24	02.31 ± 04.13	02.36 ± 08.15
NTN	F_1	47.98 ± 03.46	47.98 ± 03.46	47.98 ± 03.46	56.78 ± 03.24	56.78 ± 03.24	56.78 ± 03.24
	M%	57.68 ± 03.43	57.68 ± 03.43	57.68 ± 03.43	60.40 ± 05.45	60.40 ± 05.45	60.40 ± 05.45
	C%	06.52 ± 07.54	06.52 ± 07.54	06.55 ± 07.54	06.55 ± 07.54	06.55 ± 07.55	06.55 ± 07.55
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	35.88 ± 08.84	35.88 ± 08.84	35.88 ± 08.84	33.05 ± 05.73	33.05 ± 05.73	33.05 ± 05.73
HD	F_1	50.44 ± 00.13	50.44 ± 00.13	50.44 ± 00.13	65.43 ± 00.35	67.43 ± 00.43	65.43 ± 00.13
	M%	59.49 ± 00.03	59.49 ± 00.03	59.49 ± 00.03	67.56 ± 00.43	68.43 ± 00.43	67.56 ± 00.42
	C%	00.47 ± 00.05	00.47 ± 00.05	00.47 ± 00.05	00.14 ± 00.26	00.14 ± 00.27	00.14 ± 00.28
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	40.04 ± 00.26	40.04 ± 00.26	40.04 ± 00.26	32.30 ± 00.26	32.30 ± 00.26	32.30 ± 00.26

uncertain-membership more easily than the forests obtained without the application of the pruning strategy. However, thanks to simplification strategy, the size of the resulting forests was considerably reduced: after pruning, the average size of the ETRFs did not exceed 10 trees. Tab. 12 reports the average forest sizes[8].

5.2.3. Evaluating cases of induction

One of the most important consequences of the *credulous* behavior of ETDTs and ETRFs was the large induction rates, which represent the cases of non

[8] The sizes have been averaged over the folds and, the resulting values have been further averaged over the number of target concepts.

Table 9: Outcomes for ETRFs adopting the three heuristics for the best concept selection and the Dubois-Prade rule as a meta-learner, with and without the use of the pruning

Ontology		No simplification			simplification		
		Non-specific.	Dissonance	Confusion	Non-specific.	Dissonance	Confusion
Financial	F_1	90.89 ± 03.25	90.97 ± 03.36	91.23 ± 03.76	96.85 ± 03.25	96.85 ± 03.25	96.85 ± 03.25
	M%	93.43 ± 05.06	93.89 ± 05.16	94.03 ± 05.23	97.12 ± 03.10	97.13 ± 04.12	97.12 ± 04.15
	C%	01.07 ± 01.67	01.71 ± 02.50	00.77 ± 01.74	00.60 ± 00.03	00.54 ± 00.03	00.77 ± 01.74
	O%	03.22 ± 00.15	02.19 ± 00.15	02.90 ± 00.14	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	02.28 ± 08.13	02.31 ± 08.17	02.30 ± 08.15	02.28 ± 08.13	02.31 ± 08.17	02.30 ± 08.15
monetary	F_1	92.39 ± 04.97	94.76 ± 05.76	94.45 ± 07.15	95.57 ± 06.15	97.21 ± 05.67	97.43 ± 03.35
	M%	93.43 ± 05.06	95.89 ± 05.16	94.56 ± 04.46	96.65 ± 04.35	99.43 ± 08.13	99.55 ± 08.15
	C%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	06.57 ± 05.06	04.11 ± 05.16	01.97 ± 07.16	03.35 ± 04.35	00.57 ± 08.13	00.45 ± 08.15
DBpedia	F_1	59.89 ± 03.78	59.89 ± 03.78	50.23 ± 02.43	68.12 ± 03.24	68.12 ± 03.24	68.12 ± 03.24
	M%	63.84 ± 05.43	63.84 ± 05.43	54.46 ± 05.43	70.43 ± 03.31	70.43 ± 03.31	70.43 ± 03.31
	C%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	35.16 ± 23.30	35.16 ± 23.30	35.16 ± 23.30	29.56 ± 03.31	29.56 ± 03.31	29.56 ± 03.31

logically derivable definite classifications whose correctness requires a validation from a domain expert.

However, the number of new assertions resulting from the inductive classification models was very large, especially in the experiments with NTN, HD, and DBPedia. As a consequence, there was a drastic decrease of the F-measure as it considers such cases as label mismatches, whereas the induction rate treats them as non conflictual assertions that could be exploited in a perspective of integration and evolution of the KBs.

Devising a different strategy for tackling these cases of induction, we designed and performed new experiments, considering a modified version of the ontologies. The new versions were obtained by introducing disjointness axioms in accordance with the *strong disjointness assumption* (SDA) which states that sibling concepts in the subsumption hierarchy can be considered as disjoint [30]. In this way, the cases of individuals with uncertain-membership can be minimized or totally avoided and a ground truth with definite membership labels can

Table 10: Outcomes for ETDTs under *Strong Disjointness Assumption*

Ontology		Non-specif.	Dissonance	Confusion
NTN	F_1	92.17 ± 07.56	93.78 ± 07.43	93.78 ± 07.43
	M%	93.45 ± 07.67	94.67 ± 07.85	94.67 ± 07.85
	C%	06.55 ± 07.67	05.33 ± 07.85	05.33 ± 07.85
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
HD	F_1	94.12 ± 03.57	94.12 ± 03.57	94.12 ± 03.57
	M%	96.46 ± 04.56	96.46 ± 04.56	96.46 ± 04.56
	C%	03.54 ± 04.56	03.54 ± 04.56	03.54 ± 04.56
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
DBpedia	F_1	91.05 ± 02.43	91.05 ± 02.43	91.05 ± 02.43
	M%	92.35 ± 03.97	92.35 ± 03.97	92.35 ± 03.97
	C%	07.65 ± 03.97	07.65 ± 03.97	07.65 ± 03.97
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00

be provided to evaluate induction cases. Tab. 10 and 11 illustrate the results of the new experiments with NTN, HD and DBpedia. Note that, under the SDA, most of the cases previously classified as cases of induction were deemed as matching cases via both ETDTs and ETRFs. Again, the performance of the ETRFs overcame the one obtained through a single tree in terms of F_1, match rate and also with a decrease of the standard deviation. With the adoption of the SDA, the match rates were less biased by the parameter ϵ tuned for ETDTs and ETRFs.

5.2.4. *Experiments with ETDTs and ETRFs with Special Probabilistic BBAs*

For the sake of completeness, we tested the effectiveness of a modified version of the ETDT and ETRF models (and related algorithms) such that the BBAs in their nodes had only singletons as focal elements. To this purpose, the function computeBBA in Alg. 1 has been adapted as follows: the probability mass assigned to $m(\Omega)$ *has been proportionally distributed to the singletons* $\{-1\}$ *and* $\{+1\}$ (preserving the sum (1) of the focal elements).

Similarly to the previous experiments, Tab. 13 and 14 illustrate the outcomes of this comparison only for NTN, HD and DBpedia ontologies, where the results significantly changed w.r.t. the original versions (in the experiments with

Table 11: Outcomes for the ETRFs under *Strong Disjointness Assumption*

Ontology		No simplification			simplification		
		Non-specif.	Dissonance	Confusion	Non-specif.	Dissonance	Confusion
NTN	F_1	96.23 ± 03.13	96.32 ± 04.43	96.32 ± 04.18	95.87 ± 04.56	96.74 ± 03.85	96.74 ± 03.85
	M%	96.57 ± 04.23	96.60 ± 04.17	96.60 ± 04.17	95.87 ± 04.56	96.74 ± 03.85	96.74 ± 03.85
	C%	03.43 ± 04.23	03.40 ± 04.17	03.40 ± 04.17	04.13 ± 04.56	03.26 ± 03.85	03.26 ± 03.85
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
HD	F_1	97.23 ± 00.16	97.43 ± 00.16	97.43 ± 00.17	97.23 ± 00.16	97.43 ± 00.16	97.43 ± 00.16
	M%	98.56 ± 00.43	98.80 ± 00.45	98.70 ± 00.34	98.60 ± 00.44	98.76 ± 00.32	98.76 ± 00.33
	C%	01.44 ± 00.43	01.20 ± 00.45	01.30 ± 00.34	01.40 ± 00.44	01.24 ± 00.32	01.24 ± 00.33
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
DBpedia	F_1	99.43 ± 00.12	99.43 ± 00.12	99.43 ± 00.12	99.43 ± 00.12	99.43 ± 00.12	99.22 ± 03.14
	M%	99.20 ± 03.21	99.26 ± 03.17	99.16 ± 03.21	99.21 ± 03.13	99.21 ± 03.12	99.22 ± 03.14
	C%	00.80 ± 03.21	00.74 ± 03.17	00.84 ± 03.21	00.70 ± 03.13	00.79 ± 03.12	00.78 ± 03.14
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00

Table 12: Average size of forests (number of trees) after the pruning

Ontology	*Forest size after the pruning*		
	10 trees	20 trees	30 trees
BCO	6.76	6.43	5.76
BioPax	5.56	5.43	5.76
NTN	7.87	7.45	7.43
HD	8.43	7.65	6.56
Financial	4.34	4.43	4.43
monetary	8.44	7.65	7.42
DBpedia	8.44	7.23	7.33

the other ontologies, the results did not change because the BBAs of the trees have already singletons as focal elements). The tables report the outcomes obtained inducing ETDTs and ETRFs with the mixing rule (for pooling the BBAs in the leaf-nodes) and Dubois-Prade's rule as a meta-learner. Similar values have been obtained in the evaluation with the other rules.

Generally speaking, we noted a decay of the performance in terms of F-measure, both for ETDTs and ETRFs w.r.t. the original versions, especially in terms of (an increased) commission rate. On a close inspection of the models, we observed that the BBAs at the leaves nodes computed with the new proce-

Table 13: Outcomes for ETDTs with BBAs having singletons as focal elements

Ontology		Non-specif.	Dissonance	Confusion
NTN	F_1	73.23 ± 12.54	76.24 ± 12.43	73.15 ± 12.44
	M%	73.42 ± 11.43	77.32 ± 07.85	74.23 ± 07.85
	C%	13.32 ± 12.43	09.15 ± 04.34	09.15 ± 03.85
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	13.26 ± 08.84	13.26 ± 08.84	13.26 ± 08.84
HD	F_1	67.16 ± 13.43	67.16 ± 13.43	67.16 ± 13.43
	M%	70.01 ± 07.26	70.01 ± 07.26	70.01 ± 07.26
	C%	14.65 ± 04.56	14.65 ± 04.56	14.65 ± 04.56
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	15.33 ± 01.35	15.34 ± 01.35	15.34 ± 01.34
DBpedia	F_1	86.43 ± 03.35	86.43 ± 03.35	86.43 ± 03.35
	M%	87.85 ± 13.23	87.85 ± 13.23	87.85 ± 13.23
	C%	00.37 ± 00.30	00.37 ± 00.29	00.37 ± 00.29
	O%	00.30 ± 00.06	00.30 ± 00.06	00.30 ± 00.06
	I%	11.27 ± 08.73	11.27 ± 08.73	11.27 ± 08.73

dure tended to favor the majority class with high assigned values. When such functions are pooled through a combination rule, the final decision was strongly biased towards such class. As a consequence, the models determined a wrong membership value for the test individuals. Another remarkable difference is in the lower induction rate, that was likely due to the induction of trees in which the focal values of the BBAs, $m(\{+1\})$ and $m(\{-1\})$, located in the leaf-nodes were often (approximately) equal. Such cases represented the main source for ties, resulting in a label 0 returned. In this sense, even resorting to forests instead of single trees did not allow to considerably improve the performance: the membership assessed by one tree was further confirmed by the other trees in the forest.

5.2.5. Comparison with other inductive systems

As previously described, ETDTs and ETRFs showed a more credulous behavior w.r.t. the other learning systems used in the experiments, in particular compared to the instance-based methods and CELOE (see Tab. 15 and 16). The k-NN showed a very *cautious* behavior: the neighborhood of the test in-

Table 14: Outcomes for the ETRFs with BBAs having only singletons as focal elements

Ontology		No simplification			simplification		
		NON-SPECIF.	DISSONANCE	CONFUSION	NON-SPECIF.	DISSONANCE	CONFUSION
NTN	F_1	74.08 ± 08.15	74.08 ± 08.15	74.08 ± 08.15	75.16 ± 10.54	75.16 ± 10.54	75.16 ± 10.54
	M%	75.34 ± 09.23	75.23 ± 09.23	75.24 ± 09.24	76.87 ± 09.14	76.88 ± 09.14	76.88 ± 09.14
	C%	13.32 ± 12.21	13.43 ± 12.21	13.42 ± 12.20	13.32 ± 12.21	13.32 ± 12.21	13.32 ± 12.21
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	11.34 ± 08.76	11.34 ± 08.76	11.34 ± 08.84	09.81 ± 04.23	09.81 ± 04.23	09.81 ± 04.23
HD	F_1	73.25 ± 07.42	73.26 ± 07.42	73.25 ± 07.42	73.25 ± 07.42	73.25 ± 07.42	73.25 ± 07.42
	M%	74.32 ± 05.13	74.32 ± 05.13	74.32 ± 05.13	74.32 ± 05.13	74.32 ± 05.13	74.32 ± 05.13
	C%	10.31 ± 04.56	10.31 ± 04.56	10.31 ± 04.56	10.31 ± 04.56	10.31 ± 04.56	10.31 ± 04.56
	O%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	15.33 ± 01.35	15.34 ± 01.35	15.34 ± 01.34	15.33 ± 01.35	15.34 ± 01.35	15.34 ± 01.34
DBPEDIA	F_1	86.43 ± 03.35	86.43 ± 03.35	86.43 ± 03.35	86.43 ± 03.35	86.43 ± 03.35	86.43 ± 03.35
	M%	87.85 ± 13.23	87.85 ± 13.23	87.85 ± 13.23	87.85 ± 13.23	87.85 ± 13.23	87.85 ± 13.23
	C%	00.37 ± 00.30	00.37 ± 00.29	00.37 ± 00.29	00.37 ± 00.30	00.37 ± 00.29	00.37 ± 00.29
	O%	00.30 ± 00.06	00.30 ± 00.06	00.30 ± 00.06	00.30 ± 00.06	00.30 ± 00.06	00.30 ± 00.06
	I%	11.27 ± 08.73	11.27 ± 08.73	11.27 ± 08.73	11.27 ± 08.73	11.27 ± 08.73	11.27 ± 08.73

dividuals was often made up of uncertain individuals. This explains both the very high match rate achieved with this algorithm in the experiments with NTN and the high omission rate observed in the experiments with DBPEDIA. In the experiments with CELOE, introducing a stricter definition of negative example than the one originally adopted in [27], made the algorithm more sensitive to lack of disjointness axioms and, consequently, led to omission cases rather than commission errors. Conversely, in case of ontologies with an explicit specification of disjointness axioms, the match rate tended to be very high (in some cases close to 100%), thanks to a strategy that aims at maximizing the F-measure.

Finally, in the experiments with PERCEPTRON, we observed a drop of the match rate and an increase of commission and induction cases. On one hand, the higher commission rates were due to overfitting models, likely owing to the large number of epochs adopted in the experiments. On the other hand, the higher induction rates were due to the decision procedure adopted in the classification phase, which tended to assign a definite membership rather than an uncertain membership to test individuals.

Table 15: Outcomes for other learning systems

Ontology		TDT	TRF	κ-NN	CELOE	Perceptron
BCO	F_1	76.23 ± 03.01	84.78 ± 02.43	84.78 ± 02.43	100.0 ± 00.00	83.45 ± 12.45
	M%	80.44 ± 11.01	87.99 ± 07.85	87.83 ± 12.43	100.0 ± 00.00	86.27 ± 15.79
	C%	07.56 ± 08.08	04.32 ± 04.68	12.77 ± 04.77	00.00 ± 00.00	02.47 ± 03.70
	O%	05.04 ± 04.28	00.09 ± 00.27	00.02 ± 00.04	00.00 ± 00.00	00.00 ± 00.00
	I%	06.96 ± 05.97	07.61± 06.82	00.40 ± 00.00	00.00 ± 00.00	09.36 ± 13.96
BioPax	F_1	64.23 ± 13.26	71.43 ± 03.24	77.23 ± 03.46	100.0 ± 00.00	63.43 ± 15.46
	M%	66.63 ± 14.60	75.93 ± 17.05	75.49 ± 17.05	75.30 ± 16.23	65.30 ± 16.23
	C%	31.03 ± 12.95	22.11 ± 16.54	18.54 ± 17.80	18.74 ± 17.80	18.74 ± 17.80
	O%	00.39 ± 00.61	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	01.95 ± 07.13	01.97 ± 07.16	01.97 ± 07.16	01.97 ± 07.16	11.97 ± 05.76
NTN	F_1	63.24 ± 10.98	81.43 ± 03.35	95.23 ± 03.23	78.91 ± 08.43	96.14 ± 08.43
	M%	68.85 ± 13.23	83.42 ± 07.85	96.82 ± 03.43	83.42 ± 07.85	96.81 ± 07.46
	C%	00.37 ± 00.30	00.00 ± 00.00	00.02 ± 00.04	00.02 ± 00.04	00.02 ± 00.04
	O%	09.51 ± 07.06	13.40 ± 10.17	00.00 ± 00.00	00.02 ± 00.04	00.00 ± 00.00
	I%	21.27 ± 08.73	03.16 ± 04.65	03.16 ± 04.65	00.00 ± 00.00	03.17 ± 04.65
HD	F_1	54.23 ± 14.15	62.85 ± 10.43	66.54 ± 17.11	65.00 ± 17.63	66.42 ± 16.43
	M%	58.31 ± 14.06	67.95 ± 16.99	67.96 ± 17.00	67.95 ± 17.03	68.00 ± 16.98
	C%	00.44 ± 00.47	00.02 ± 00.05	00.01 ± 00.05	00.02 ± 00.05	00.02 ± 00.05
	O%	05.51 ± 01.81	06.38 ± 02.03	06.38 ± 02.03	06.38 ± 02.03	06.38 ± 02.03
	I%	35.74 ± 15.90	25.61 ± 18.98	25.61 ± 18.98	25.61 ± 18.98	25.59 ± 18.98

5.2.6. *Efficiency of the methods*

A final remark is related to the efficiency of the proposed approaches. Considering Tab. 17 it can be noted that the averaged run-times of the ETDT and ETRF models spanned from less than 35s to almost 13000s. The efficiency of the solutions proposed in this paper depends on the size of training sets and the number of concepts and roles contained in the signature of the knowledge bases. While the former affected the performance in terms of the number of tests to be performed in the training/test phase, which was intensively used by ETDTs and ETRFs, the latter affected the generation of the complex concept descriptions installed into the nodes. Also, the pruning procedure employed for optimizing the ensemble models represented a further complexity source in the training phase but simpler models brought an increased efficiency in the prediction phase. Overall, the efficiency of the new models in both training and

Table 16: Outcomes for other learning systems

Ontology		TDT	TRF	κ-NN	CELOE	Perceptron
	F_1	66.23 ± 36.01	96.23 ± 02.56	96.23 ± 02.56	99.12 ± 00.73	74.32 ± 00.87
	M%	67.06 ± 36.09	96.70 ± 00.48	96.70 ± 00.65	99.70 ± 00.68	79.50 ± 00.68
Financial	C%	00.00 ± 00.00	02.00 ± 03.43	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	O%	32.94 ± 36.09	00.00 ± 00.60	00.30 ± 00.68	00.30 ± 00.68	00.00 ± 00.00
	I%	00.00 ± 00.00	01.30 ± 00.50	00.00 ± 00.00	00.00 ± 00.00	20.50 ± 00.68
	F_1	66.12 ± 15.23	94.13 ± 07.74	100.0 ± 00.00	100.0± 00.00	65.43 ± 15.96
	M%	68.93 ± 15.87	94.53 ± 07.68	100.0 ± 00.00	100.0 ± 00.00	68.93 ± 15.87
monetary	C%	06.14 ± 07.20	05.47 ± 07.68	00.00 ± 00.00	00.00 ± 00.00	06.14 ± 07.20
	O%	16.94 ± 09.74	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	I%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00
	F_1	08.12 ± 01.21	08.12 ± 01.21	08.12 ± 01.21	56.13 ± 20.43	58.12 ± 15.47
	M%	10.86 ± 01.69	10.86 ± 01.69	10.86 ± 01.69	58.84 ± 20.35	63.93 ± 15.07
DBpedia	C%	43.12 ± 00.57	43.12 ± 00.57	43.12 ± 00.57	30.28 ± 20.10	25.18 ± 14.48
	O%	46.02 ± 01.64	46.02 ± 01.69	46.02 ± 01.69	00.00 ± 00.00	00.00 ± 00.00
	I%	00.00 ± 00.00	00.00 ± 00.00	00.00 ± 00.00	10.86 ± 01.69	10.86 ± 01.69

test phase is comparable to the one of TDTs and TRFs and also close to the average execution time with the k-NN. Indeed, one of the main bottlenecks of the lazy learning approach was related to the (exhaustive) search of the nearest neighbors for each test individual. Moreover, we noted that the evidential models were more efficient than perceptron. In this case, the run-times of Perceptron were affected mainly by the inefficiency of the training phase in which, for each epoch, all the training examples are processed to determine the coefficients of the classification model.

6. Related Work

The knowledge made available in a decentralized form across the Semantic Web is often contradictory, imprecise and incomplete [31]. Machine learning can be exploited for setting up methods providing alternative forms of reasoning. In this work, we specifically focused on the task of assessing the membership of an individual with respect to a target concept. This problem has been largely investigated in the literature and various approximate classification models have

Table 17: Ranges of average run-time (training / test) per experiment (s)

Ontology	ETDT [min, max]	ETRF [min, max]	TDT [min, max]	TRF [min, max]	k-NN [min, max]	CELOE [min, max]	Perceptron [min, max]
BCO	[35, 40]	[120, 453]	[35, 40]	[120, 452]	[65, 87]	[15, 37]	[480,520]
BioPax	[67, 87]	[123, 456]	[70, 103]	[145, 523]	[83, 245]	[23, 60]	[345, 725]
NTN	[432, 785]	[876, 1256]	[578, 876]	[914, 1243]	[123, 456]	[12,65]	[765, 1343]
HD	[446, 879]	[1245, 1278]	[446, 895]	[1567, 1568]	[123, 456]	[12,65]	[1234, 1456]
Financial	[1845, 2567]	[18765,42345]	[1845, 2567]	[18765,42345]	[12456,12876]	[87,247]	[24569,56797]
monetary	[2476, 4587]	[3687,45890]	[2444, 4598]	[3687,45890]	[14876,15321]	[124,256]	[49872,58931]
DBpedia	[3211, 4237]	[12345,12789]	[3211, 4237]	[12345,12789]	[8769,14321]	[87,231]	[23456,60432]

been proposed [1].

Non-parametric methods are among the most common solutions. Among the others, the k-nearest neighbor procedure [1] (also employed in the experiments) and the *reduced Coulomb energy network* [32] have been proposed. Both approaches exploit a language-independent distance measure between individuals in a DL knowledge base. Such a metric is computed based on a set of projection functions that express the behavior of an individual w.r.t. a set of concepts (treated as logic *features*). Essentially the aim is selecting prototypical individuals and classifying unseen ones on the ground of the similarity w.r.t. the closest prototypes, the neighborhood, that in the latter case is mediated by a network model (similar to the *radial basis function networks* [33]). Other related solutions are based on the explicit adaptation of *kernel methods*. For instance, in the evaluation, we used the kernel *perceptron* [28] adopting a kernel function that is closely related to the distance measure adopted by the classifiers described above [29].

Other solutions stem from *concept learning* algorithms devised in ILP to solve a closely related problem. The goal is to obtain an explicit intensional definition (a concept description in terms of the language bias of choice) describing the available examples that should be general enough to account also for unseen instances. Various algorithms of this kind have been proposed, e.g. DL-Foil [34], CELOE [27] and the mentioned method for the induction of terminological decision trees [3]. The latter extend decision trees for multi-relational representations (such as first-order logic fragments [4, 5] and *selection graphs* [35, 36]) towards SW representations. A related approach, based on models called *Semantic Decision Trees*, has been proposed in [37]. Although they are indeed quite similar to the mentioned TDTs [3], their empirical evaluation did not compare these models and it was limited to very small knowledge bases. All these approaches are based on the use of a refinement operator in order to progressively build such description(s). However, such concept learning methods often do not provide a strategy for representing uncertainty, although various efforts have been devoted to investigate the effectiveness of models combining multi-relational representation languages and uncertainty, in the context of *Statistical Relational Learning* [38] or *Probabilistic Inductive Logic Programming* [39]. Among the existing solutions it is possible to mention *Bayesian Logic Programs* [40] and *Markov Logic Networks* (MLNs) [41]. Focusing on MLNs, a *domain closure* assumption is required thus diverting from the open-world semantics of the FOL fragments adopted as standard representations in the SW context [42]. However, the assumptions for inducing MLNs can be relaxed by using an EM algorithm to learn from incomplete data [43]. In this perspective, a recent work [44] has proposed a functional-gradient boosting algorithm based on EM in order to learn, under the OWA, the structure and the parameters of the models simultaneously.

The need to circumvent the exponential growth of the model (and hence of the number of parameters) required by the groundings justified the works on *approximation methods* [45] and *lifted inference techniques* [46, 47] Alternatively, *tensor models* have also been proposed [48, 49] although the limitations in terms of scalability of such complex statistical models remains. That is why currently *representation learning* approaches [50] have attracted the attention of the community. They trade the focus on the mere relational structure of the rich SW KBs with a low rank representation which is more manageable with standard geometric-statistical approaches.

Note that, due to the different expressiveness of the languages underpinning ILP and SRL methods w.r.t. those for the SW representations, the application of such solutions is not straightforward. This problem has been considered since the early works that apply machine learning methods to DL knowledge bases. For instance,in [51], the authors have shown that there may be an exponential blowup in knowledge base size and there may be some formulae without a counterpart in DLs. Further issues have been discussed in [52] where the author argues that *ad-hoc* solutions may avoid both exploring a larger search space (represented by the set of all possible Horn clauses) and the limitations of the complex reasoning services required by logic programming.

In order to better represent the inherent uncertainty related to the specific semantics of the SW knowledge bases, the DST [21] offers an interesting alternative, which explicitly considers the ignorance deriving from the inherent incompleteness of the KBs and the availability of further evidence. Additionally, the DST has been successfully integrated in various machine learning algorithms to enhance the predictiveness of the models. For instance, DST primitives have been integrated in the k-nearest neighbor algorithm [53], where each example in the neighborhood is considered as a distinct source of evidence in favor of a class that is subsequently combined through Dempster's rule [21]. In the SW context, a DL-compliant version of this approach has been proposed for solving the class-membership prediction problem [54]. The DST has been integrated also with algorithms for learning neural networks [55] and decision trees [56]. Indeed, the latter inspired our idea of evolving TDTs towards the ETDTs [7]. Differently from the original version of such a model (which is intended for a propositional representation), the induction of an ETDT is guided by the non-specificity measure whereas the original model considers also conflictual evidences. In this paper we have extended our investigation considering further total uncertainty measures.

The DST has been employed in the context of ensemble learning for pooling the prediction coming from the weak learners [13, 15]. Various ensemble combination methods resort to *decision templates*, which are obtained by fitting, for each classifier against each class, a mean vector (called *reference vector*). When these models are employed, predictions are typically made by computing the similarity between a decision profile of an unknown instance with the decision templates. Unlike such approaches, the decision procedure employed with the ETRFs combines the predictions returned in the form of BBAs. In this sense, this procedure is similar to the one proposed in [57]: each classifier returns a BBA that is combined by the meta-learner implementing a combination rule. Again, ETRFs work on multi-relational representation language, similarly to their original version, namely the *Terminological Random Forests* [10]. This ensemble model, which represents a subtype of the First Order Logic Random Forests [12] that is compliant with DLs, has been devised to tackle the problem of class-imbalance in datasets drawn from Semantic Web knowledge bases (and to overcome the limits of other solutions, such as those adopting sole sampling methods [26]), which is an issue that had not been tackled before. A random forest model for Semantic Web knowledge bases has been also proposed in [58] but, unlike TRFs and ETRFs, the solution exploits only atomic concepts as features.

One of the contributions of this paper concerns the adoption of a pruning procedure for ETRFs, which mitigates some problems derived from the use of many classifiers (e.g. the inefficiency in the prediction step) and can determine a good forest size per learning problem. In general, the problem of determining such number is still an open issue: even in the case of simpler representation languages (attribute-value and propositional logic), there were only few works that propose solutions which are often based on the use of statistical tests (e.g. *McNeimar's test*) [59]. Instead, this number is a parameter whose value is typically intended as user-provided [60]. Only in a recent work regarding the application of random forests on data streams [61], the authors argued that the ideal number of classifiers is strictly related to the number of class labels of the dataset.

7. Conclusion and Extensions

We have proposed and extended a framework for inducing evidential terminological decision trees and random forests, as developments of the terminological decision trees and random forests, devised as solutions of the problem of class-membership prediction for Semantic Web knowledge bases. Following the lessons learned with previous versions, the new models tackle various shortcomings affecting the quality of the models, especially the cases of uncertain classification and imbalanced datasets due to the inherent incompleteness of the knowledge bases of interest. The resulting models combine predictions that are represented as basic belief functions rather than votes, exploiting evidence combination rules proposed in the context of the Dempster-Shafer Theory for making the final decision. In addition, for evidential terminological random forests, a strategy for optimizing the ensemble has been proposed.

Extensive experiments have been performed to assess the validity of the proposed models, also considering datasets drawn from various Web ontologies, varying conditions and parameter settings, and in comparison with other inductive models and learning strategies. The experiments have shown how the proposed classification model can achieve a better predictiveness than the previous versions of terminological decision trees and random forest. In various cases, the results are better than the other learning systems.Moreover, the models tended to assign a definite membership yielding to induce a large number of non logically derivable assertions whose correctness was assessed under the Strong Disjointness Assumption [30].

Besides, the predictiveness of the evidential terminological decision trees was found not to depend on the rule adopted for combining evidence while the predictiveness of evidential terminological random forests was not affected by the choice of either the forest size or the sampling rate The standard deviation is also lower than the one observed with the original TRFs. The evaluation showed that the simplification procedure proposed to optimize the ensemble favors the prediction of uncertain membership .

In the future, we plan to extend the method along various directions. One regards considering an explicit *semi-supervised* learning approach for DL classifiers so to assign a definite membership to the uncertain examples. In this case, it could be possible to devise solutions inspired from multi-view learning approaches [62]. In addition, it can be interesting to investigate the effectiveness of kernels derived from evidential random forests, as proposed in [63].

Further ensemble techniques and novel rules for combining the answers of the weak learners could be employed. For example, weak learners can be induced from subsets of training instances generated by means of a procedure based on cross-validation rather than sampling with replacement. Further investigations may concern the application of strategies aiming at the optimization of the ensembles during the induction of the classifier rather than *ex post*, i.e. after the training phase has been completed.

Finally, the (ensemble) methods could be naturally parallelized and the resulting decision procedure based on induced models could be made available as a service i.e. a non-standard inference service to complement standard query answering or reasoning services. In this perspective, using specific frameworks such as *Apache Spark*[9] or GPUs may be an interesting alternative to be considered.

References

[1] A. Rettinger, U. Lösch, V. Tresp, C. d'Amato, N. Fanizzi, Mining the Semantic Web - Statistical learning for next generation knowledge bases, Data Min. Knowl. Discov. 24 (2012) 613–662.

[2] C. d'Amato, N. Fanizzi, B. Fazzinga, G. Gottlob, T. Lukasiewicz, Ontology-based semantic search on the web and its combination with the power of inductive reasoning, Ann. Math. Artif. Intell. 65 (2012) 83–121.

[3] N. Fanizzi, C. d'Amato, F. Esposito, Induction of concepts in web ontologies through terminological decision trees, in: J. Balcázar, et al. (Eds.),

[9] http:/spark.apache.org

Proceedings of ECML/PKDD2010, volume 6321 of *LNAI*, Springer, 2010, pp. 442–457.

[4] H. Blockeel, Top-down induction of first order logical decision trees, Ph.D. thesis, Department of Computer Science, Katholieke Universiteit Leuven, 1998.

[5] H. Blockeel, L. De Raedt, Top-down induction of first-order logical decision trees, Artif. Intell. 101 (1998) 285–297.

[6] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Schneider (Eds.), The Description Logic Handbook, Cambridge University Press, 2nd edition, 2007.

[7] G. Rizzo, C. d'Amato, N. Fanizzi, On the effectiveness of evidence-based terminological decision trees, in: F. Esposito, et al. (Eds.), Proceedings of ISMIS 2015, volume 9384, Springer, 2015, pp. 139–149.

[8] F. Smarandache, D. Han, A. Martin, Comparative study of contradiction measures in the theory of belief functions, in: Proceedings of FUSION 2012, pp. 271–277.

[9] H. He, Y. Ma, Imbalanced Learning: Foundations, Algorithms, and Applications, Wiley-IEEE Press, 1st edition, 2013.

[10] G. Rizzo, C. d'Amato, N. Fanizzi, F. Esposito, Tackling the class-imbalance learning problem in semantic web knowledge bases, in: K. Janowicz, et al. (Eds.), Proceedings of EKAW 2014, volume 8876 of *LNCS*, Springer, 2014, pp. 453–468.

[11] L. Breiman, Random forests, Machine Learning 45 (2001) 5–32.

[12] A. V. Assche, C. Vens, H. Blockeel, S. Dzeroski, First order random forests: Learning relational classifiers with complex aggregates, Machine Learning 64 (2006) 149–182.

[13] L. Kuncheva, A theoretical study on six classifier fusion strategies, Pattern Analysis and Machine Intelligence, IEEE Transactions on 24 (2002) 281–286.

[14] L. Xu, A. Krzyzak, C. Suen, Methods of combining multiple classifiers and their applications to handwriting recognition, Systems, Man and Cybernetics, IEEE Transactions on 22 (1992) 418–435.

[15] G. Rogova, Combining the results of several neural network classifiers, in: R. Yager, L. Liu (Eds.), Classic Works of the Dempster-Shafer Theory of Belief Functions, volume 219 of *Studies in Fuzziness and Soft Computing*, Springer, 2008, pp. 683–692.

[16] X.-C. Yin, C. Yang, H.-W. Hao, Learning to diversify via weighted kernels for classifier ensemble, CoRR abs/1406.1167 (2014).

[17] Y. Bi, J. Guan, D. Bell, The combination of multiple classifiers using an evidential reasoning approach, Artificial Intelligence 172 (2008) 1731 – 1751.

[18] G. Rizzo, C. d'Amato, N. Fanizzi, F. Esposito, Inductive classification through evidence-based models and their ensembles, in: F. Gandon, et al. (Eds.), Proceedings of ESWC 2015, volume 9088 of *LNCS*, Springer, 2015, pp. 418–433.

[19] K. Sentz, S. Ferson, Combination of evidence in Dempster-Shafer theory, Technical Report, SANDIA, SAND2002-0835, 2002.

[20] J. Klir, Uncertainty and Information, Wiley, 2006.

[21] G. Shafer, A Mathematical Theory of Evidence, Princeton University Press, Princeton, 1976.

[22] D. Dubois, H. Prade, On the combination of evidence in various mathematical frameworks, in: J. Flamm, T. Luisi (Eds.), Reliability Data Collection and Analysis, volume 3 of *Eurocourses*, Springer, 1992, pp. 213–241.

[23] D. Dubois, H. Prade, A note on measures of specificity for fuzzy sets, International Journal of General Systems 10 (1985) 279–283.

[24] U. Höhle, A general theory of fuzzy plausibility measures, Journal of Mathematical Analysis and Applications 127 (1987) 346 – 364.

[25] R. R. Yager, Entropy and specificity in a mathematical theory of evidence, in: R. Yager, L. Liu (Eds.), Classic Works of the Dempster-Shafer Theory of Belief Functions, volume 219 of *Studies in Fuzziness and Soft Computing*, Springer, 2008, pp. 291–310.

[26] H. He, E. A. Garcia, Learning from imbalanced data, IEEE Trans. on Knowl. and Data Eng. 21 (2009) 1263–1284.

[27] J. Lehmann, S. Auer, L. Bühmann, S. Tramp, Class expression learning for ontology engineering., J. Web Sem. (2011) 71–81.

[28] J. Shawe-Taylor, N. Cristianini, Kernel Methods for Pattern Analysis, Cambridge University Press, New York, NY, USA, 2004.

[29] N. Fanizzi, C. d'Amato, F. Esposito, Induction of robust classifiers for web ontologies through kernel machines, J. Web Sem. 11 (2012) 1–13.

[30] S. Schlobach, Debugging and semantic clarification by pinpointing, in: Gómez-Pérez, et al. (Eds.), Proceedings of ESWC 2005, volume 3532 of *LNCS*, Springer, 2005, pp. 226–240.

[31] K. Laskey, P. Costa, M. Kokar, T. Martin, T. Lukasiewicz, Uncertainty Reasoning for the World Wide Web, Technical Report, 2008.

[32] N. Fanizzi, C. d'Amato, F. Esposito, ReduCE: A reduced coulomb energy network method for approximate classification, in: L. Aroyo, et al. (Eds.), Proceedings of ESWC 2009, volume 5554 of *LNCS*, Springer, 2009, pp. 323–337.

[33] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Springer, 2 edition, 2009.

[34] N. Fanizzi, C. d'Amato, F. Esposito, DL-FOIL Concept Learning in Description Logics, in: Proceedings of ILP 2008, LNAI, Springer, 2008, pp. 107–121.

[35] V. Tripathy, A comparative study of multi-relational decision tree learning algorithm, International Journal of Scientific & Technology Research 2 (2013).

[36] H. A. Leiva, A multi-relational decision tree learning algorithm, Master's thesis, Department of Computer Science. Iowa State University, 2002.

[37] D. Jeon, W. Kim, Development of semantic decision tree, in: The 3rd International Conference on Data Mining and Intelligent Information Technology Applications, pp. 28–34.

[38] L. Getoor, B. Taskar, Introduction to Statistical Relational Learning (Adaptive Computation and Machine Learning), The MIT Press, 2007.

[39] L. De Raedt, P. Frasconi, K. Kersting, S. Muggleton (Eds.), Probabilistic Inductive Logic Programming: Theory and Applications, Springer, 2008.

[40] K. Kersting, L. De Raedt, Basic principles of learning bayesian logic programs, in: L. De Raedt, et al. (Eds.), Probabilistic Inductive Logic Programming: Theory and Applications, volume 4911 of *LNCS*, Springer, 2008, pp. 189–221.

[41] M. Richardson, P. Domingos, Markov logic networks, Mach. Learn. 62 (2006) 107–136.

[42] P. M. Domingos, D. Lowd, S. Kok, H. Poon, M. Richardson, P. Singla, Just add weights: Markov logic for the semantic web, in: P. C. G. da Costa, et al. (Eds.), Uncertainty Reasoning for the Semantic Web I, ISWC International Workshops, URSW 2005-2007, Revised Selected and Invited Papers, volume 5327 of *Lecture Notes in Computer Science*, Springer, 2008, pp. 1–25.

[43] P. Domingos, S. Kok, D. Lowd, H. Poon, M. Richardson, P. Singla, Markov Logic, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 92–117.

[44] T. Khot, S. Natarajan, K. Kersting, J. Shavlik, Gradient-based boosting for statistical relational learning: the markov logic network and missing data cases, Machine Learning 100 (2015) 75–100.

[45] S. Sarkhel, D. Venugopal, T. A. Pham, P. Singla, V. Gogate, Scalable training of markov logic networks using approximate counting, in: D. Schuurmans, M. P. Wellman (Eds.), Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA., AAAI Press, 2016, pp. 1067–1073.

[46] B. Ahmadi, K. Kersting, M. Mladenov, S. Natarajan, Exploiting symmetries for scaling loopy belief propagation and relational training, Machine Learning 92 (2013) 91–132.

[47] A. Kimmig, L. Mihalkova, L. Getoor, Lifted graphical models: A survey, Mach. Learn. 99 (2015) 1–45.

[48] M. Nickel, V. Tresp, H. Kriegel, A three-way model for collective learning on multi-relational data, in: L. Getoor, T. Scheffer (Eds.), Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011, Omnipress, 2011, pp. 809–816.

[49] M. Nickel, Tensor factorization for relational learning, Ph.D. thesis, Ludwig Maximilians University Munich, 2013.

[50] A. Bordes, N. Usunier, A. García-Durán, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: C. J. C. Burges, L. Bottou, Z. Ghahramani, K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States., pp. 2787–2795.

[51] L. Badea, S. H. Nienhuys-Cheng, A Refinement Operator for Description Logics, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 40–59.

[52] J. Lehmann, Learning OWL Class Expressions, Ph.D. thesis, University of Leipzig, 2010. PhD in Computer Science, supervisors: Prof. Klaus-Peter Fähnrich, Dr. Sören Auer.

[53] T. Denoeux, A k-nearest neighbor classification rule based on dempster-shafer theory, IEEE Transactions on Systems, Man, and Cybernetics 25 (1995) 804–813.

[54] G. Rizzo, C. d'Amato, N. Fanizzi, F. Esposito, Assertion prediction with ontologies through evidence combination, in: F. Bobillo, et al. (Eds.), Uncertainty Reasoning for the Semantic Web II, volume 7123 of *LNAI*, Springer, 2013, pp. 282–299.

[55] T. Denoeux, A neural network classifier based on dempster-shafer theory., IEEE Transactions on Systems, Man, and Cybernetics, Part A 30 (2000) 131–150.

[56] N. Sutton-Charani, S. Destercke, T. Denoeux, Classification trees based on belief functions, in: T. Denoeux, M.-H. Masson (Eds.), Belief Functions: Theory and Applications, volume 164 of *Advances in Intelligent and Soft Computing*, Springer, 2012, pp. 77–84.

[57] Y. Bi, J. Guan, D. Bell, The combination of multiple classifiers using an evidential reasoning approach., Artif. Intell. 172 (2008) 1731–1751.

[58] D. Jeon, W. Kim, Random forest algorithm for linked data using a parallel processing environment, IEICE Transactions on Information and Systems E98.D (2015) 372–380.

[59] P. Latinne, O. Debeir, C. Decaestecker, Limiting the number of trees in random forests, in: J. Kittler, F. Roli (Eds.), Multiple Classifier Systems, Second International Workshop, MCS 2001 Cambridge, UK, July 2-4, 2001,

Proceedings, volume 2096 of *Lecture Notes in Computer Science*, Springer, 2001, pp. 178–187.

[60] T. M. Oshiro, P. S. Perez, J. A. Baranauskas, How many trees in a random forest?, in: Proceedings of the 8th International Conference on Machine Learning and Data Mining in Pattern Recognition, MLDM'12, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 154–168.

[61] H. R. Bonab, F. Can, A theoretical framework on the ideal number of classifiers for online ensembles in data streams, in: S. Mukhopadhyay, et al. (Eds.), Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016, ACM, 2016, pp. 2053–2056.

[62] S. Sun, A survey of multi-view machine learning, Neural Computing and Applications 23 (2013) 2031–2038.

[63] A. Davies, Z. Ghahramani, The random forest kernel and other kernels for big data from random partitions., CoRR abs/1402.4293 (2014).