An application of Dempster - Shafer model for reasoning with uncertain knowledge on the Web

Loukia Karanikola
Department of Informatics and Telecommunications
University of Athens
Ilissia University Campus
Greece
Email: lkaranik@di.uoa.gr

Isambo Karali
Department of Informatics and Telecommunications
University of Athens
Ilissia University Campus
Greece
Email: izambo@di.uoa.gr

Sally McClean
School of Computing and Information Engineering
University of Ulster
Coleraine
N. Ireland
U.K
Email: si.mcclean@ulster.ac.uk

***Abstract*—The Semantic Web vision introduces an automated way of information processing. A basic characteristic of Web information is that it is not always clear-cut and a *deficiency degree*, i.e. uncertainty and vagueness, describes it. Any representation scheme provided for web information should take into account this characteristic and a reasoning method for this scheme should also be defined. Our approach faces these issues, i.e. the representation scheme and the reasoning algorithm. In order to show how our method performs, an implementation is outlined.**

I. Introduction

The content of Web information has been designed for human consumption, i.e. it is human oriented. The evolution of search engines gave a boost at the popularity of WWW, but at the same time made it necessary for the existence of a Web (or Web information) suitable for machines (or agents). All related problems can be summed up in the phrase: "*The meaning of Web Content is not machine-accessible* [1]". On the other hand, the Semantic Web was the vision of Tim Berners-Lee who stated: "*I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web, the content, links, and transactions between people and computers. A "Semantic Web", which should make this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The intelligent agents people have touted for ages will finally materialize.*"[2]. The Semantic Web will contribute in the evolution of many web applications [1], e.g.:

- Knowledge management
- Business-to-Computer
- Electronic Commerce
- Wikis

So, the Semantic Web vision introduces the notion of machine-oriented information. This information comes as a result of data existing in various web sources. Information extraction from these sources can be very difficult in many cases. Reliability, ambiguity or incompleteness issues are the usual problems considering Web information, resulting in deficient knowledge. Any method that represents machine-oriented information should provide a well-defined description of imperfect knowledge and a reasoning method suitable for this representation should be used.

In this paper, we present an approach for reasoning for the Semantic Web when deficient knowledge is taken into account. Our approach is based on Dempster - Shafer model of reasoning under uncertainty [3]. Dempster - Shafer has been chosen as it is suitable for cases when incomplete information exist. Moreover, the Dempster's rule of combination serves as a method of integrating information from various sources which is often required in distributed environments.

This paper is organized as follows: In Section 2, an overview of the basic semantic web notions is given. In Section 3, the *deficient knowledge* characteristics are introduced along with a description of some reasoning methods. Also the basic characteristics of Dempster-Shafer model are given. In Section 4, our approach is introduced describing a representation scheme and a reasoning method that can operate on uncertain knowledge. In Sections 5 and 6, the ontology definition and the reasoning method are presented. In Section 7 an example of our method is given. In Section 8 a *hotel metaclassifier* application is presented, as an application area of our method. Finally, we give our conclusions and a plan for future work.

II. Semantic Web notions

One of the most important notions of Semantic Web is that of the "*agent*". As it is referred in [1], a semantic web agent "*will receive some tasks and preferences from a person, seek information from web sources, communicate with other agents, compare information about user requirements and preferences, select certain choices, and give answers to the users*". It seems that the role of an agent actually demands a decision making mechanism, which in turn presupposes a method of handling uncertainty and vagueness tasks. Another basic concept of Semantic Web and in general semantic applications is that of "*ontology*". Generally, an ontology "*is an explicit and formal specification of a conceptualization*"[1]. We can consider that an ontology consists of [4]:

1) Types of entities that describe a specific domain
2) Properties of those entities

Moreover, the emergence of Semantic Web incorporates a lot of technologies, which are described in what we call a *semantic* *web stack* [1] [1]. *Ontologies* and *logic* are the most significant among these technologies.

A. Ontologies

Roughly speaking, an ontology is a *conceptualization of a domain*. Ontologies are defined through the OWL (Web Ontology Language) family, the ontology language recommended by W3C [6]. There exist three categories of OWL, OWL-Full, OWL-DL and OWL-Lite. OWL-DL and OWL-Lite are based on description logics, which is a logic-based knowledge representation formalism for modeling a domain in terms of concepts (classes), roles (properties / relations) and individuals [7], [8]. Ontologies represent the semantics of the domain (in the case of SW the semantics of the source).

B. Logic

As it is referred in [1] logic is the "*foundation of knowledge representation*". One of the main characteristics of logic is the proof systems that exist that provide a way to reason in order to inference new knowledge. Considering Semantic Web the reasoning methods exist are suitable for *crisp logic*, i.e statements that are *true/false*. As we will see next, in cases where uncertainty and vagueness exist, these methods are not applicable.

III. Deficient knowledge representation and reasoning - Current Approaches

A. Deficient knowledge

As it is referred in the introductory section, web information is characterized as *deficient*. More precisely, deficient knowledge can be divided in two categories:

- *Uncertainty*: As it is referred in [4] "*uncertainty is ubiquitous*". Uncertainty refers to situations when information incompleteness exist in order to decide about the truthness of a fact.
- *Vagueness*: It refers to situations when imprecise meaning considering concepts exist, or as it is referred in [9] the *clarity* property does not exist.

A good example of uncertainty and vagueness is given in [10], where the word "*degree*" is used to describe both uncertainty and vagueness measurements, but with different meaning. For example,

1) "*To some degree birds fly*" (uncertainty)
2) "*To some degree Jim is blond and young*" (vagueness)

Moreover, uncertainty can be divided into [11]:

1) *Aleatory Uncertainty*: It results from the fact that a system can behave in random ways. In that case, uncertainty is represented by relative frequencies [12].
2) *Epistemic Uncertainty*: It results from the lack of knowledge about a system. In that case, uncertainty reflects subjective assessments of likelihood.

The main idea of the aforementioned notions is that, generally, people do not use the probability measure to describe ignorance. The latter is the most usual cause of uncertainty on the Semantic Web [12].

B. Uncertainty notions

Considering uncertainty, the following notions are representations include the notion of possible worlds that are called "states" or *elementary outcomes* [12]. Another important notion is that of event. As it is mentioned in [12] an *event* is a set of possible worlds. Having defined *possible world* and *event*, the uncertainty of an agent demands the definition of the following notions [12]:

- W: Set of possible worlds
- W' $\subseteq$ W: Subset that the agent considers possible, i.e. a qualitative measure of uncertainty
- U $\bigcap$ W' $\neq \emptyset$: U is possible (according to the agent)
- W' $\subseteq$ U: The agent knows U

C. Uncertainty and Dempster-Shafer

In problems considering the Semantic Web, uncertainty comes as a result of ignorance, which in turn is due to *incomplete information*. In those cases, the classical notion of probability is not suitable for the following reasons [12]:

1) Probability is not as good at representing ignorance.
2) An agent cannot always define probabilities for all sets of possible worlds.
3) In some cases, the computational effort demanded for probability definition, might be prohibitive.

The basic notion of Dempster-Shafer theory is the *belief function* (or support function [12]). This theory attaches likelihood to events. The belief function can be described as "*a measure of evidence that supports an event*". Dempster - Shafer theory introduces the following definitions:

1) *frame of discernment* W: It is defined as the set of different and mutually exclusive events. Another characteristic of the frame of discernment is that the propositions contained are "*exhaustive*"[3]. In general, all uncertainty representation methods consider what we call "a set of possible wordls", i.e. all the possible answers to a question of interest. For example, when we toss a die and the question is: "What is the outcome?". Then, the answer is in the set $\{1, 2, 3, 4, 5, 6\}$. This set is the set of possible outcomes (W).
2) *power set of* W: In addition, the theory considers 2^W as the set of all subsets of W, i.e. the power set of W. For example, if $W = \{e, e'\}$, where e' means "not e", then 2^W is $\{\emptyset, \{e\}, \{e'\}, \{e, e'\}\}$.

On each element U_i of the set 2^W , the following functions are defined:

1) *Basic probability assignment or mass function, bpa*: It assigns to each element of 2^W a number between 0 and 1.

[1] Also in [5] the semantic web architecture is regarded as "two-towers" rather than a stack

2) *Belief function, bel*: It is defined as the sum of all bpa of subsets of U_i.
3) *Plausibility function, pl*: It is the sum of all bpa of $U_j \in 2^W$, such that $U_i \cap U_j \neq \emptyset$

There exists a one-to-one relation between the belief function and the basic probability assignment, i.e. "to every mass function there corresponds a unique belief function and conversely for every belief function there corresponds a unique mass function"[3]. In [3] the equation for computing the mass function from belief function is given.

Thus, Dempster-Shafer is a theory that concludes degrees of belief for a statement (question of interest) based on probabilities of other statements. The notion of belief functions provide "a non-Bayesian way of using mathematical probability to quantify subjective judgments"[13]. As mentioned in [13], the theory of belief functions is based on two ideas:

- Idea of obtaining degrees of belief for one question from subjective probabilities for a related question
- Dempster's rule of combination of such degrees of belief when they are based on independent items of evidence

In general, Dempster-Shafer theory considers [14]:

- Combination of evidence
- Data fusion

The Dempster-Shafer model provides us with the ability to "assess 'belief' on some space Y on which the existence of probability measure is acknowledged, but not precisely known in that the probability is known for some of its subsets, not for all of them"[11]. The Dempster's combination rule provides a way to tackle inconsistency handling described above. One of its prerequisites is the independence of statements considered. One other main reason for the use of Dempster-Shafer theory is the fact that it performs well in situations of "epistemic uncertainty"[11]. In order to perform combination of evidence two aspects should be considered[11]:

1) The type of evidence
2) Inconsistency (or conflicting) evidence handling

D. Probabilistic Ontologies

In [10], a probabilistic ontology is defined as a twofold notion:

- Considering terminological knowledge, i.e concepts and roles, this is extended into terminological probabilistic knowledge
- Considering assertional knowledge, i.e. instances of concepts and roles, this is extended into assertional probabilistic knowledge

These probabilistic ontologies are called into existence in order to fulfill the following tasks [10]:

- Representation of terminological and assertional probabilistic knowledge
- Information retrieval
- Ontology matching
- Probabilistic data integration

E. Implemented technologies

Having taken into account the aforementioned concepts, the current approaches toward uncertainty and vagueness representation are the following:

1) Fuzzy OWL: In [15], an extension of OWL with fuzzy sets is introduced. This extension results in fuzzy facts, as well as the semantics of this extended language. In addition, the f-OWL Axiom is defined, as well as the fuzzy ontology concept. Also, a membership degree is defined for each fact that belongs to a specific concept.

2) Probabilistic RDF: In [16] a probabilistic generalization of RDF, pRDF, is implemented. A pRDF instance is defined as an extension of RDF triples with unconditional probability distributions over a set of possible values. In addition, a pRDF schema is a "probabilistic quadruple of the form $(s, rdfs : subClassOf, O, \delta)$", where s is a class, O is another class and δ denotes a probability distribution over O. The combination of pRDF instance and pRDF schema defines a pRDF theory.

3) Probabilistic OWL: In [4] and [17], the probabilistic ontology is introduced, as an extension of the ontology notion. More specifically, a Bayesian framework for probabilistic ontologies is defined, which leads into an extension of Web Ontology Language. The extended language is called PR-OWL.

4) Possibilistic Description Logics: In [18], a possibilistic approach of Description logic is implemented through the KAON2 reasoner. This approach uses possibilistic logic in order to represent uncertainty.

F. Reasoning about deficient knowledge

As it is referred in II.B in situations when crisp logic cannot be preserved, i.e. when a *degree* notion as a result of uncertainty or vagueness exist, a reasoning method for this knowledge should also be provided. In general, there exist two categories of reasoning, *monotonic* and *nonmonotonic* [19]. In order to perform reasoning under uncertainty, the classical notion of Knowledge Base should be extended into a *Probabilistic Knowledge Base*. As it is described in [10], a probabilistic knowledge base is characterized by the following:

- Finite nonempty set of basic events $\Phi = \{p1, \ldots, pn\}$
- Event ϕ: Boolean combination of basic events
- Logical constraint $\psi \Leftarrow \phi$: events ψ and ϕ: ϕ implies ψ
- Conditional constraint $(\psi|\phi)$[l,u]: events ψ and ϕ , and l,u $\in$ [0,1]: "conditional probability of ψ given ϕ is in [l,u]". Furthermore, as it is stated in [20] the conditional constraints can be divided into:
    1) Strict conditional constraints: They always hold, as for example "All sparrows are birds"
    2) Defeasible conditional constraints: Generally, they hold, but under certain circumstances

they might not. For example, "Generally, people use their right hand for writing" So, defeasible constraints represent weaker connections that can be defeated.

- Probabilistic knowledge base KB=(L,P):
  - finite set of logical constraints L
  - finite set of conditional constraints P

In our approach, as we will show, the notations described above are used. In order to perform reasoning in probabilistic knowledge bases, the following steps are followed [20]:

- A nonempty set, called At, of basic events is defined. This set is called also *set of basic propositions*.
- Set of classical formulas: It is the closure of $At \bigcup \{\bot, \top\}$
- The strict and defeasible constraints are defined as ($\psi|\phi$)[l,u] and ($\psi \parallel \phi$)[l,u] respectively.
- A probabilistic default theory is defined as $T = (P, D)$, where P is a finite set of strict conditional constraints and D is a finite set of defeasible conditional constraints.
- Set of strict probabilistic formulas: It is defined as the closure of the set of all strict conditional statements under the Boolean operators $\wedge$ and $\neg$.
- Set of probabilistic formulas: It is defined as the closure of the set of all conditional statements under the Boolean operators $\wedge$ and $\neg$.
- A possible world is a truth assignment $I : At \rightarrow \{true, false\}$
- I_{At} denotes the set of all possible worlds
- Probabilistic interpretation - Pr: It is a probability function that assigns to each possible world I a number [0,1]. The probabilistic interpretation suggests a probability ordering among possible worlds. For example, when tossing a coin that is fair, we consider the possible world $head$ equally likely to the possible world $tails$ (that's why we consider $Pr(head) = Pr(tails = 0.5)$).
- Satisfaction of probabilistic formula: A probabilistic interpretation Pr satisfies (or Pr is a model of) a probabilistic formula (strict or defeasible) iff $Pr(\psi \mid \phi)\epsilon[l, u]$.
- Verification of a default: A Pr verifies a default $Pr(\psi \parallel \phi)\epsilon[l, u]$ if $Pr(\phi) = 1$ and Pr satisfies $Pr(\psi \parallel \phi)\epsilon[l, u]$. Also, a set of defaults D *tolerates* a default d *under* a set of strict conditional constraints P iff $P \bigcup D$ has a model, i.e. a Pr that satisfies it that also satisfies d. If such model does not exist, it is stated that D is *in conflict* with d.
- Default ranking σ: It is defined on a set of defaults D as: $\sigma : d \rightarrow \{0...1\}$, for each $d \in D$. It is admissible with $T = (P, D)$ iff each $D' \subseteq D$ that is under P in conflict with a default d, has a $d' : \sigma(d') < \sigma(d)$. If such an admissible default ranking exists, then T is σ *-consistent*.
- Probability ranking κ : It is a mapping that assigns a ranking number to each probability interpretation, i.e $\kappa : P_r \rightarrow \{0...1\} \bigcup \infty$, where $\kappa(P_r) = 0$ for at least one P_r. Also for any satisfiable formula F, $\kappa(F) = min(\kappa(P_r) \mid P_r \models F)$. If F is not satisfiable, then $\kappa(F) = \infty$. Also, if $\kappa(\neg F) = \infty$ then κ is *admissible* with F. Considering a default formula D, then κ is *admissible with* D iff $\kappa((\phi \top)[1, 1])$ and $\kappa((\phi \mid \top) \wedge [1, 1](\psi \mid \phi)[l, u]) < \kappa((\phi \mid \top)[1, 1] \wedge \neg(\psi \mid \phi)[l, u])$

If we consider a set of strict and defeasible conditional constraints, as well as some evidence, then in [20] three reasoning methods are described that are suitable for *non-crisp logic*:

1) z-entailment: This method considers a $T = (P, D)$ and defines the *z-partition* of the set of defaults D as $(D_0, ..., D_n)$. Each D_i contains a set of defaults that are tolerated under P. Each default $d \in D$ exists only in one D_i. After, the default ranking z on D is defined as well as the probability ranking κ^z that are admissible with T. The κ^z suggests an ordering on P_r, and it is stated that if $\kappa(P_r) < \kappa(P_{r'})$ then P_r is z-preferable to $P_{r'}$. Also, among all models M of a set of probabilistic formulas F, a P_r is z-minimal iff: $\kappa^z(P_r) < \kappa^{z(P_{r'})}, \forall P_{r'} \in M$. Also, if KB is a knowledge base, then a strict probabilistic formula F is a *z-consequence* of KB iff each *z-minimal model* of $P \bigcup KB$ satisfies F. Finally, if $F = (\psi \mid \phi)[l, u]$ and l, u are the infimum and supremum of all $P_r(\psi \mid \phi)$, such that P_r z-minimal model and nonnegative, respectively, then $F = (\psi \mid \phi)[l, u]$ is a *tight z-consequence* of KB.

2) lexicographic entailment: This method defines a *z-partition* $(D_0, ..., D_n)$ on D, in a way that a P_r is lexicographic preferable to $P_{r'}$ iff there exists a $j \in 0, ..., n$ such that the number of $d \in D_j$ that P_r satisfies is greater than the number of $P_{r'}$ and also for all $t < j$ the number of $d \in D_t$ that P_r satisfies is equal to that of $P_{r'}$.

3) conditional entailment: Here, *priority orderings* on D are defined, denoted as $\prec$. The *admissible* and $\prec$-consistent notions are defined in an analogous way as for default ranking σ. Also the notion of $\prec$-preferable exists, as well as the $\prec$-minimal model. The *$\prec$-entailment* and *strict $\prec$-entailment* are defined in an analogous way as for z-entailment.

The aforementioned approaches extend the classical notions of entailment in order to take into account conditional constraints. For performing entailment, these methods incorporate generic knowledge, i.e. statistical knowledge, which comes under the term "objective knowledge", and evidence, which comes under the term "subjective knowledge". The concept of these methods is the following:

- A set of strict and defeasible constraints is considered. This set comes under the term "Probabilistic default theory - T"
- Some evidence is also given, which constitutes a "knowledge base - KB".
- The set of strict conditional constraints, as well as the evidence, should always be satisfied, i.e. there should be a probabilistic interpretation Pr model of them, while performing any entailment method.

- The above probabilistic interpretation should also be a model of a subset of the set of defaults in a way that any member of this set is not in conflict with any other member under the set of strict conditional constraints and the evidence.
- Also, they ignore irrelevant information, show property inheritance to globally nonexceptional subclasses, and respect the principle of specificity.

Considering inconsistency handling, those methods do not always entail "intuitively" expected conclusions. There are situations where some methods have better performance than the others. Moreover, z-entailment and lexicographic entailment consider the principle of specificity and conditional entailment entails ignorance. One other characteristic of these approaches is that in order to perform reasoning, they entail conclusions from subjective knowledge using objective knowledge.

IV. Uncertain knowledge representation and reasoning - ImpOnto system

As it is referred in section IV, Dempster-Shafer model is suitable in situations where epistemic uncertainty exist, i.e. when uncertainty is due to lack of knowledge. Next, we present an approach that uses Dempster-Shafer theory in order to represent uncertainty and also a reasoning method that uses Dempster's rule of combination. In order to apply our method, we use the characteristics of knowledge bases described in paragraph II-E. Also, as we will show next our method uses two well known metrics used in statistics and probability theory, mean value and standard deviation, in order to combine information from various sources. The main idea behind our method is the following:

Combine two (or more) of the entailment methods described in section II-E in order to get a result that is a combination of the results of the results of the entailment methods.

A. Representation of ontological knowledge

In order to represent uncertain knowledge, an ontology has been defined. Our goal is to represent statements that are aligned with Dempster-Shafer model and hence have the following form:

"An element of the power set of the possible world set is true with belief bel, plausibility pl and basic probability assignment bpa"

Our uncertainty ontology applies a set of concepts, as we will see next, some of them are:

1) **Possible World**: It represents the set of possible states, in other words, the frame of discernment.
2) **Power Set of the set "Possible World"**: It is the power set of the above set.
3) **Agent**: An agent makes the aforementioned statements.

Considering reasoning, our method combines the entailment results from the different entailment methods described in II.E, i.e z-entailment, lexicographic entailment and z-entailment and derives a combined result. So, our method still preserves:

1) *Ignores irrelevant information*
2) *Applies property inheritance to globally nonexceptional subclasses*
3) *Applies the principle of specificity*

Following the conventions described above, we have the following definitions:

1) **Strict conditional constraint**: It has the form $(c|b)[l,u]$ and represent generic knowledge that always hold
2) **Defeasible (default) conditional constraint**: It has the form $(c \parallel b)[l,u]$ and represent weaker generic knowledge that can be defeated.
3) **Probabilistic default theory**: (P,D) that represents generic knowledge plus evidence

As we stated above, the goal of a reasoning method is to derive conclusions. More precisely, we want to know if an "*event e is true*". So, our frame of discernment is defined through the truthness of the event e, i.e.

- **Frame of discernment**: W={e,e'}
- **Power Set**: 2^W is $\{\emptyset, \{e\}, \{e'\}, \{e,e'\}\}$

All the entailment methods described in [20] derive a conclusion as a *strict probabilistic formula*, i.e as $(\psi \mid \top)[k,l]$. In order to combine these conclusions through Dempster-Shafer rule of combination, a single number should be defined as the bpa for each statement (constraint). There are three ways we can define the bpa through probability intervals:

1) $bpa = l$ (pessimistic approach)
2) $bpa = u$ (optimistic approach)
3) $bpa = \frac{(l+u)}{2}$ (middle approach)

We decide to use the 3rd approach due to the fact that it seems intuitively better, hence we have the following definition:

Define the bpa of a statement as: $bpa = \frac{(l+u)}{2}$

Also, it should be noticed that if we consider l and u as $belief$ and $plausibility$ function respectively, then the equation given in [3] can be used in order to compute the bpa.

B. Multi-value estimations

The approaches described above can be applied in situations where we have an interval, i.e. lower and higher value. Besides that, there are cases where we have more values that a higher and lower ones. For example, in a hotel rating web site, like *www.booking.com*, there exist for each hotel a list of ratings. In those situations, we have to take into account all the information given, i.e. all the values. A scalar measure of the values is the *mean value*. The mean value can constitute a belief measure, but in some situations it can be *deceitful*. For example, if there exist two estimations $e1$ and $e2$ with values 0 and 1 respectively, then the mean value of them is 0.5. Suppose now that we have two other estimations $e1'$ and $e2'$ with values 0.45 and 0.55 respectively, then the mean value is also 0.5. Although the two pairs of estimations have the same mean value, our belief is stronger in the second case. The reason is that in the second pair the estimations are very *close with respect to their values*. We can say that our *credibility* for the sources of information, and hence our belief augments as differences among estimation values get smaller. The measure that describes the variation among the values is the standard deviation. The intuitive rule is the following:

The smaller the standard deviation the higher our credibility is

Having taken this rule into account, our belief (or bpa as we talk about basic elements) for a set of estimations can be defined as follows:

$$bpa = |meanvalue - standarddeviation|$$

C. Combination

Having defined all the necessary Dempster Shafer functions, we can perform Dempster's Rule of Combination considering W, 2^W and for all bpa_s for computing a single bpa:

The bpa of the event e is $bpa_{1,2,\ldots,n}(e)$

The combined bpa, using Dempster's combination rule, for event e is computed as follows:

$$bpa_{1,2}(e) = \frac{\sum_{x,y \in W : x \cap y = e} bpa_1(x) \times bpa_2(y)}{1 - \sum_{x,y \in W : x \cap y = \emptyset} bpa_1(x) \times bpa_2(y)}$$

Now, the belief function for the event *e*, bel(e), has the same value as the $bpa_{1,2,\ldots,n}$ (e), since e constitutes a basic element (it does not have subsets):

Define $bel(e)$ ***as*** $bpa_{1,2,\ldots,n}$ ***(e) (entailment)***

V. Ontology definition

In order to apply our approach described in the previous section, we implemented an ontology using the OWL API in Java language. Our ontology consists of the following concepts (fig. 1):

1) **Possible World Concept**: An instance $i \in PossibleWorldConcept$ if i is a member of the Possible World set.
2) **Power Set Concept**: An instance $i \in PowerSetConcept$ if i is a member of the Power Set.
3) **Agent Concept**: An instance $i \in AgentConcept$ if i has been characterized as an Agent.
4) **Result Concept**: An instance $i \in ResultConcept$ if i has been derived as a conclusion from our Dempster - Shafer reasoning process.
5) **HasFor Concept**: As it is stated above, an agent makes statements about power set elements. For example, *The element el of the powerset set has bel b*. Considering our implementation, this statement actually joins an $agent$ instance, a $powerset$ instance and a bel. As the relations should be binary, this triple can only be represented through the insertion of a new concept. This is the role of the HasFor concept.

Also, our ontology contains the following relations among concepts:

1) **hasBel relation**: It is a belief function that is defined for the elements of the power set
2) **hasPl relation**: It is a plausibility function that is defined for the elements of the power set
3) **hasBpa relation**: It is a basic probability assignment that is defined for the elements of the Power Set
4) **hasfor relation**: It connects an agent individual (instance) with an element of the Power Set and a bpa through the HasFor Concept.
5) **hasPID relation**: It assigns unique id number (pid) to each element of the Power Set
6) **hasID relation**: It assigns a unique number (id) to each agent individual (instance)

Our uncertainty ontology as well as the reasoning method are implemented in Java language using the OWL API. Our application name is called *ImpOnto* (Fig.2). The individuals of the *Possible World* class and the *Agents* class are inserted by the user, where the individuals of the *Power Set* are computed automatically. As it is stated above, the bpas are the mean

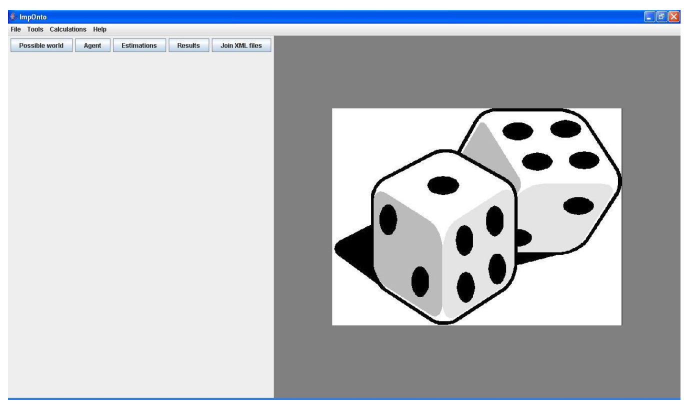

Fig. 1. ImpOnto

values of the probability intervals. After interval insertion the ontology hierarchy tree has the following form (Fig.3): As it is shown in figure, on the top of the hierarchy is our application name, i.e. *ImpOnto*. The next step of hierarchy contains the concept names *Possible World*, *Power Set*, *Agent* and *Has For*, as described above. The instances of these concepts are shown in the next level.

VI. Reasoning method algorithm

As it is referred in section IV, our goal is to combine different entailment methods in order to derive a conclusion. As we consider each entailment method as an *independent source*, we can use Dempster's rule of combination in order to combine the results for two (or more) methods. The results, which are part of the ontology hierarchy, are actually belief numbers for each powerset element. We have to notice that all the entailment methods in [20] derive a conclusion as a strict probabilistic formula, i.e. in the form:

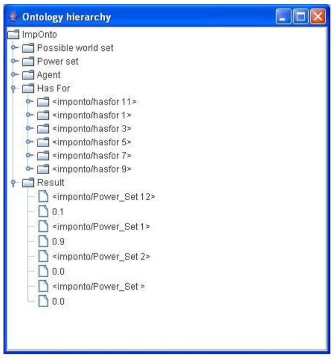

Fig. 2. Ontology hierarchy

$$(\phi \mid \psi)[l, u]$$

On the other hand, Dempster's rule of combination demands a single value (in the form of bpa) as an input. So, the intervals are mapped first into a single value, using the middle approach described above. Our reasoning method consists of the following steps:

1) *Mapping process*:
   - *Interval estimations*: $bpa = \frac{(l+u)}{2}$
   - *Multi-value estimations*: $bpa = |meanvalue - standarddeviation|$

   The *bpa* of the element e, as we talk about power sets of the form $\{\emptyset, \{e\}, \{e'\}, \{e, e'\}\}$, is equal to $bel(e)$.
2) *Combination process*: The bpa_s from step one are combined through Dempster's rule of combination in order to derive a conclusion

VII. IMPONTO EXAMPLE FOR INTERVAL ESTIMATION

In order to illustrate how our method works, we consider the following probabilistic default theory as described in [20]:
P={(bird|penguin)[1,1]}
D={(fly||bird)[0.95,1],
(fly||penguin)[0,0.05]
, (easy_to_see||yellow)[0.95,1.0]} We also have the following evidence:
KB={(penguin∧yellow|⊤)[1,1]} The entailment methods described in [20] derive the following conclusions:

- z-entailment: $easy\_to\_see \mid \top[0, 1]$
- lexicographic entailment: $easy\_to\_see \mid \top[0.95, 1]$
- conditional entailment: $easy\_to\_see \mid \top[0.95, 1]$

Our goal is to derive a combined conclusion for two (or more) of the entailment methods. For our example we choose the first and the last, i.e. z and *conditional* entailment. We consider the following frame of discernment: $X = \{easy\_to\_see, \neg easy\_to\_see\}$. Then, $2^X = \{\emptyset, \{easy\_to\_see\}, \{\neg easy\_to\_see\}, \{easy\_to\_see, \neg easy\_to\_see\}\}$. The two entailment methods are considered "independent pieces of evidence" since each method is not based on the other. The independence is a necessary precondition for applying Dempster Shafer rule of combination. Also, in order to apply combination rule, the basic probability assignments are defined as follows:

- $m_z(\emptyset) = 0$
  $m_z(\{easy\_to\_see\}) = \frac{0+1}{2} = 0.5$
  $m_z(\{easy\_to\_see, \neg easy\_to\_see\}) = 1 - 0.5 = 0.5$
- $m_{cond}(\emptyset) = 0$
  $m_{cond}(\{easy\_to\_see\}) = \frac{0.95+1}{2} = 0.975$
  $m_{cond}(\{easy\_to\_see, \neg easy\_to\_see\}) = 1 - 0.975 = 0.025$

The Dempster's rule of combination produces the following result: $m_{z,cond}(\{easy\_to\_see\}) = 0.9875$ As we observe, our belief about *easy to see* is greater than the initial beliefs, which is intuitively correct, since belief$_z$ increases initial belief$_{cond}$ (and vice versa).

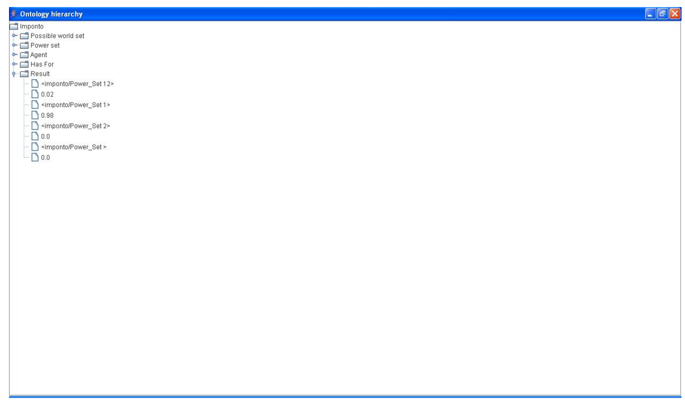

Fig. 3. Entailment method result

VIII. A METACLASSIFIER SYSTEM

Next we give an application of our system in the field of classification. More precisely, as we will show in the next example, Dempster's rule of combination can be used as a *metaclassifier* for deriving a classification result from different classifiers. In our example we consider two well-known hotel classifiers, *www.booking.com* and *www.tripadvisor.co.uk*. Each classifier assigns each hotel a rating. Our application takes as input two urls considering hotel estimations, one for each classifier. If we consider the set of ratings provided by each classifier as *multi-value estimations* and each classifier as an independent source, then we can use our method in order to derive a single-value estimation. More precisely, the ratings are the result of various reviews from past hotel residents. These reviews can be represented in a tree structure: The first step is to compute the *mean value* of each set of estimations. Moreover, the mean value itself does not provide always a useful information. For example, if there exist too many *proposed* ratings as well as too many *not proposed* ones, we can say that it is not possible to derive a conclusion for the hotel. On the other hand, if we have most of the ratings close to the mean value, then we can say that the hotel *approaches*

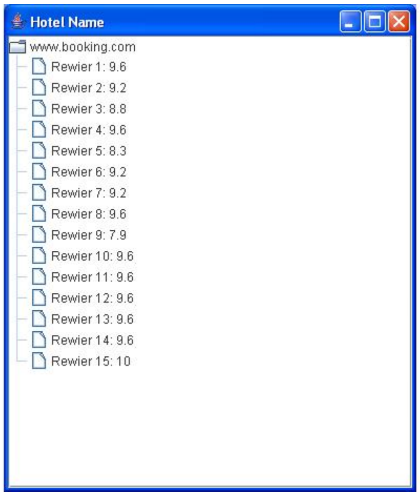

Fig. 4. Hotel ratings

a mean value rating. In order to tackle this problem we compute the standard deviation of the values. So, the belief value is computed as follows:

$$bpa = |meanvalue - standarddeviation|,$$

i.e. the standard deviation is a *measure of the credibility* of the ratings set. Our method takes as input these multi-value estimations of the two classifiers and returns a *belief number* about *how much proposed the hotel is*. In order to do this the following steps are implemented:

1) The ratings are mapped into a number [0,1]
2) The mean value and standard deviation for each set of estimations is computed. In our example, we have two hotel rating web sites, so we have two sets of ratings.
3) The bpa is $|meanvalue - standarddeviation|$
4) The frame of discernment $\{\emptyset, \{proposed\}, \{\neg proposed\}, \{proposed, \neg proposed\}\}$
5) Dempster's rule of combination is performed in order to get a combined result. As it is shown in fig.9 the results have the form *imponto/PowerSet number*, $estimation$, where $number$ defines a powerset element, e.g. in our example the element 1 defines $proposed$, and $estimation$ is the resulted belief for the specific element. So, *imponto/PowerSet 1*, 0.1 means that the *hotel is proposed with belief 0.1*.

In order to test our system a list of hotels that exist in *www.booking.com* and *www.tripadvisor.co.uk* was selected. Following are the results of our system:

Our system uses the Dempster's Rule of Combination in order to augment the belief measure about *how proposed the hotel is*, i.e. as a positive score and at the same time the standard deviation as a *penalty*, i.e. as a negative score for our belief. Both measures serve as a way to have a *belief measure* or a *bpa measure*[2] that is more intuitive and hence more realistic.

[2]In our example $bel(proposed) = bpa(proposed)$, as $proposed$ is a basic element

TABLE I. HOTEL RATINGS

Hotel name	Imponto Rating
Rixos the Palm Dubai	0.75
Holiday Inn Express Dubai Airport	0.84
Number One Tower Suites	0.66
Desert Palm	0.64
Gloria Hotel	0.81
Sheraton Deira Hotel Dubai	0.78
Rihab Rotana - Dubai	0.76
Akas-Inn Hotel Apartment	0.71
Ocean View Hotel	0.66
Chelsea Tower Hotel Apartments	0.77

IX. CONCLUSION

In this paper, we have defined and implemented an ontology that represents uncertain knowledge based on Dempster-Shafer theory. Moreover, a reasoning method has been defined suitable for combining estimations that have either *interval* or *multi-value* form. A metaclassifier system has been presented as an application of our method and finally real world data have been used to test our system. As a next step we consider an ontology representation scheme and a reasoning method that captures vague notions as well. For example, in [21], by defining the categories as fuzzy sets and the categorization score as *degree of truth* we can have and a fuzzy extension of our system.

X. ACKNOWLEDGEMENTS

This research has been co-financed by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program Education and Life- long Learning of the National Strategic Reference Framework (NSRF) Research Funding Pro- gram: THALES. Investing in knowledge society through the European Social Fund. Project code: MIS 380153.

REFERENCES

[1] G. Antoniou and F. V. Harmelen, *A Semantic Web Primer*, 2nd ed. MIT Press, 2008.

[2] T. B. Lee, *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web*. Harper San Francisco, 2000.

[3] N. Wilson, "Algorithms for Dempster-Shafer theory," in *Algorithms for Uncertainty and Defeasible Reasoning*. Kluwer Academic Publishers, 2000, pp. 421–475.

[4] P. C. G. da Costa, "Bayesian Semantics for the Semantic Web," Ph.D. dissertation, Bayesian Semantics for the Semantic Web, 2005.

[5] I. Horrocks, B. Parsia, P. F. Patel-Schneider, and J. A. Hendler, "Semantic web architecture: Stack or two towers?" in *PPSWR*, 2005, pp. 37–41.

[6] http://www.w3.org/TR/owl-features/.

[7] I. Horrocks, "Owl: A description logic based ontology language," in *CP*, 2005, pp. 5–8.

[8] F. Baader, I. Horrocks, and U. Sattler, "Description Logics," in *Handbook of Knowledge Representation*, F. van Harmelen, V. Lifschitz, and B. Porter, Eds. Elsevier, 2008, ch. 3, pp. 135–180.

[9] A. Nikolov and V. Uren, "Using the Dempster-Shafer theory of evidence to resolve ABox inconsistencies."

[10] T. Lukasiewicz and U. Straccia, "Managing Uncertainty and Vagueness in Semantic Web Languages," in *AAAI-07*. AAAI-07, 2007.

[11] K. Sentz and S. Ferson, "Combination of evidence in Dempster-Shafer theory," Tech. Rep., 2002.

[12] J. Y. Halpern, *Reasoning about Uncertainty*. MIT Press, 2005.

[13] G. Shafer, "Belief Functions," in *Readings in Uncertain Reasoning*. Morgan Kaufmann Publishers, 1990.

[14] L. A. Zadeh, “A Simple View of the Dempster-Shafer Theory of Evidence and its Implication for the Rule of Combination,” *AI Magazine*, vol. 7, pp. 85–90, 1986.

[15] G. Stoilos, G. Stamou, V. Tzouvaras, J. Z. Pan, and I. Horrocks, “Fuzzy OWL: Uncertainty and the Semantic Web,” in *PROC. OF THE INTER. WORK. ON OWL-ED05*, 2005.

[16] O. Urdea, V. Subrahmanian, and Z. Majkic, “Probabilistic RDF,” in *In Proceedings IRI-2006*, 2006.

[17] P. C. G. Costa, K. B. Laskey, and K. J. Laskey, “PR-OWL: A Bayesian Ontology Language for the Semantic Web,” in *Center for Technology-Enhanced Learning, University of Missouri-Rolla*, 2003.

[18] G. Qi, J. Z. Pan, and Q. J. A, “Possibilistic Extension of description logics,” in *In Proc. of DL07*, 2007.

[19] D. Nute, “Defeasible Logic,” in *Handbook of Logic in Artificial Intelligence and Logic Programming*. Oxford University Press, 2001, pp. 353–395.

[20] T. Lukasiewicz, “Probabilistic default reasoning with strict and defeasible conditional constraints,” Institut fur Informationssysteme, Tech. Rep., 2000.

[21] L. Karanikola, A. Katsaris, and I. Karali, “Web Page Classification Using Rules and Heuristics,” in *Artificial Intelligence and Applications (AIA 2009)*, 2009, pp. 42–47.