Semantic Web Technologies

Lecture 7: Ontology engineering: uncertainty and vagueness

Maria Keet
email: `keet -AT- inf.unibz.it`
home: `http://www.meteck.org`
blog:
`http://keet.wordpress.com/category/computer-science/72010-semwebtech/`

KRDB Research Centre
Free University of Bozen-Bolzano, Italy

14 December 2009

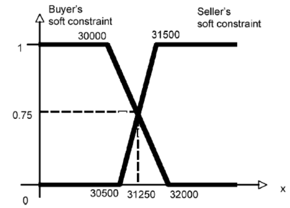
Outline

Examples

- Information Retrieval: To which degree is a Web site, a Web page, a text passage, an image region, a video segment, . . . relevant to my information need?[1]
- Matchmaking: To which degree does an object match my requirements? e.g., your budget is about 20.000 euro to buy a car, then to which degree does a cars price of 20.500 euro match your budget?
- Ontology alignment: To which degree do two concepts of two ontologies represent the same thing, or are disjoint, or are overlapping?
- Classifying ripe apples or "the set of all individuals that mostly buy low calorie food"

[1] some of the following slides are taken from Umberto Straccia's AAAI'07 tutorial [http://gaia.isti.cnr.it/∼straccia/download/papers/VANCOUVER07/VANCOUVER07.pdf]

- A car seller sells an Audi TT for 31500 euro (catalog price)
- A buyer is looking for a sports-car, but wants to to pay not more than around 30000 euro
- Classical DLs: the problem relies on the crisp conditions on price
- More fine grained approach (as usual in negotiation): consider prices as vague constraints (fuzzy sets)
  - Seller would sell above 31500 euro, but can go down to 30500
  - The buyer prefers to spend less than 30000 euro, but can go up to 32000 euro
  - Highest degree of matching is 0.75; The car may be sold at 31250 euro



- Problems: what and how to incorporate such vague or uncertain knowledge in OWL and its reasoners?
- Solutions:
  - i. probabilistic, possibilistic, fuzzy, rough extensions to the language
  - ii. for reasoning: transform back into OWL and use standard reasoner or develop your own one
- Usage, among others:
  - Information retrieval (e.g., top-k retrieval)
  - classifying patients (e.g., patients that are possibly septic have properties: infection and [temperature $>$ 38C OR temperature $<$ 36C, respiratory rate $>$ 20 breaths/minute OR PaCO2 $<$ 32 mmHg, etc])
  - Recommender systems (user preferences etc.)
  - Matchmaking in web services

Uncertainty and vagueness

- **Uncertainty**: statements are true or false, but due to lack of knowledge we can only estimate to which probability / possibility / necessity degree they are true or false
  - E.g.: a bird flies or does not fly. The probability / possibility / necessity degree that it flies is 0.83
- **Vagueness**: statements involve concepts for which there is no exact definition, such as tall, small, close, far, cheap, expensive. true to some degree, taken from a truth space
  - E.g., "Hotel Verdi is close to the train station to degree 0.83"
- Uncertainty *and* Vagueness: "It is *probable* to degree 0.83 that it will be *hot* tomorrow"
- Imperfect information covers notions such as uncertainty, vagueness, contradiction, incompleteness, imprecision

Probabilistic logic: Syntax

- Finite nonempty set of basic events $\Phi = \{p_1, \ldots, p_n\}$, with $n \geq 1$
- *Events*: every element of $\Phi \cup \{\bot, \top\}$ is an event; if ϕ and ψ are events, then so are $\neg\phi$, $(\phi \wedge \psi)$, $(\phi \vee \psi)$, and $(\phi \rightarrow \psi)$
- A *probabilistic formula* is an expression of the form $\phi \geq l$, with $l \in \mathbb{R}$ from the unit interval $[0, 1]$ (note that $\neg\phi \geq 1 - u$ encodes ϕ is true with probability at most u)
- *Conditional constraint* $(\psi \mid \phi)[l, u]$: events ψ and ϕ, and $l, u \in [0, 1]$, which denotes "the conditional probability of ψ given ϕ is in $[l, u]$"
- Probabilistic knowledge base $KB = (\mathcal{L}, \mathcal{P})$:
  - finite set of logical constraints $\mathcal{L}$
  - finite set of conditional constraints $\mathcal{P}$

Probabilistic logic: Semantics

- A world I associates with every basic event in Φ a binary truth value, and extend I by induction to all events as usual
- $\mathcal{I}_\Phi$ is the (finite) set of all worlds for Φ
- A world I satisfies an event ϕ (or: I is a model of ϕ), denoted $I \models \phi$, iff $I(\phi) = true$
- Probabilistic interpretation Pr: probability function on $\mathcal{I}_\Phi$ s.t. all $Pr(I)$ with $I \in \mathcal{I}_\Phi$ sum up to 1
- $Pr(\phi)$ is the sum of all $Pr(I)$ such that $I \in \mathcal{I}_\Phi$ and $I \models \phi$
- $Pr(\psi \mid \phi)$: if $Pr(\phi) > 0$, then $Pr(\psi \mid \phi) = \frac{Pr(\psi \wedge \phi)}{Pr(\phi)}$

Probabilistic logic: satisfiability and entailment

- A probabilistic interpretation *Pr satisfies* a probabilistic formula $\phi \geq l$ (i.e., $Pr \models \phi \geq l$) iff $Pr(\phi) \geq l$
- *Pr* satisfies a probabilistic *KB* iff *Pr* satisfies all $F \in KB$
- *KB* is satisfiable iff a model of *KB* exists
- A probabilistic formula *F* is a *logical consequence* of *KB* (denoted $KB \models F$) iff every model of *KB* satisfies *F*
- $\phi \geq l$ is a *tight logical consequence* of *KB* iff *l* is the infimum[2] of $Pr(\phi)$ subject to all models *Pr* of *KB* (the latter is equivalent to $l = sup\{r \mid KB \models \phi \geq r\}$)[3]

[2] the infimum of a subset of some set is the greatest element (not necessarily in the subset) that is less than or equal to all elements of the subset; greatest lower bound.

[3] the supremum (sup) of a subset S of a partially ordered set T is the least element of T that is greater than or equal to each element of S; least upper bound.

Probabilistic RDF, OWL, and DLs

- P-$\mathcal{SHOQ}(D)$, P-$\mathcal{SHOIN}(D)$ (by T. Lukasiewicz)
  - uses the notion of a conditional constraint
  - semantics is based on the notion of lexicographic entailment in probabilistic default reasoning
  - probabilistic TBox and ABox
  - interprets TBox and ABox probabilistic knowledge as statistical knowledge and as degrees of belief about instances of concepts and roles, respectively
  - allows for deriving both statistical knowledge and degrees of belief
  - allows for expressing default knowledge about concepts
- PR-OWL (by da Costa and Laskey)
  - Probabilistic semantics based on multi-entity Bayesian networks
- And others with Bayesian networks, with DLs, covering various permutations of probabilistic KR&R added to different languages (see references in Straccia, 2008)

Use of Probabilistic Ontologies

- Representation of terminological and assertional probabilistic knowledge (e.g., in the medical domain or at the stock exchange market)
- Information retrieval, for an increased recall
- Ontology matching
- Probabilistic data integration, especially for handling ambiguous and controversial pieces of information

Possibilistic logic introduction

- Syntactically, we now use possibilistic formulas to constrain the necessities and possibilities of propositional events
- Semantically, we now have possibility distributions on worlds, each of which associates with every event a unique possibility and a unique necessity
- Differently from the probability of an event (sum of the probabilities of all worlds that satisfy that event), the possibility of an event is the *maximum of the possibilities* of all worlds that satisfy the event
- Possibilistic logic useful for encoding user preferences, since possibility measures can be viewed as rankings (on worlds or also objects) along an ordinal scale
- While reasoning in probabilistic logic generally requires to solve linear optimization problems, reasoning in possibilistic logic does not and thus can generally be done with less computational effort

Possibilistic logic: Syntax and Semantics

- Possibilistic formulas have the form P$\phi \geq l$ or N$\phi \geq l$, with ϕ event, $l \in \mathbb{R}$ from [0, 1], **P**ossibly, and **N**ecessarly. e.g.:
  - P*snow_today* ≥ 0.7 encodes that it will snow today is possible to degree 0.7
  - N*mother* $\rightarrow$ *female* ≥ 1 says that a mother is necessarily female
- A *possibilistic formula* is a pair (ϕ, α) consisting of a classical logic formula ϕ and a degree α expressing certainty or priority (which also can be considered as possibility degree of ϕ)
- A *possibilistic knowledge base KB* is a finite set of possibilistic formulas, of the form $KB = \{(\phi_i, \alpha_i) : i = 1 \dots n\}$
- A *possibilistic interpretation* is a mapping $\pi : \mathcal{I}_\Phi \rightarrow [0, 1]$
- $\pi(I)$ is the degree to which world I is possible
  - every world I such that $\pi(I) = 0$ is impossible
  - every world I such that $\pi(I) = 1$ is totally possible
  - π is normalized iff $\pi(I) = 1$ for some $I \in \mathcal{I}_\Phi$

cont'd

- The possibility of a ϕ in a π defined by $Poss(\phi) = max\{\pi(I) \mid I \in \mathcal{I}_\Phi, I \models \phi\}$ 'possibility of ϕ is evaluated in the most possible world where ϕ is true'
- $Nec(\phi) = 1 - Poss(\neg\phi)$ 'to what extent ϕ is certainly true'
- A π *satisfies* a possibilistic formula P$\phi \geq l$ (resp., N$\phi \geq l$), or π is a model of P$\phi \geq l$ (resp. N$\phi \geq l$), denoted $\pi \models$ P$\phi \geq l$ (resp. $\pi \models$ N$\phi \geq l$), iff $Poss(\phi) \geq l$ (resp. $Nec(\phi) \geq l$)
- A possibilistic knowledge base is consistent iff its classical base is consistent

cont'd

- The *inconsistency degree* of *KB*, denoted $Inc(KB)$, is defined as $Inc(KB) = max\{\alpha_i : KB_{\geq \alpha_i} is\ inconsistent\}$
- There are two possible definitions of inference in possibilistic logic:
  - A formula ϕ is said to be a *plausible consequence* of *KB*, denoted by $KB \vdash_P \phi$, iff $KB_{> Inc(KB)} \vdash \phi$
  - A formula ϕ is said to be a *possibilistic consequence* of *KB* to degree α, denoted by $KB \vdash_\pi (\phi, \alpha)$, iff the following conditions hold: (1) $KB_{\geq \alpha}$ is consistent, (2) $KB_{\geq \alpha} \vdash \phi$, and (3) $\forall \beta > \alpha$, $KB_{\geq \beta} \nvdash \phi$
- Inference services: instance checking (plausible instance of C), plausible subsmption, instance checking with necessity degree, subsumption with necessity degree

Possibilistic ontologies

- Add it to an arbitrary DL language (including any of the DL-based OWL languages)
- E.g. Qi, Pan, Ji in DL'07, supposedly with basics implemented in KAON2
- Possibilistic generalization of $\mathcal{ALC}$ for information retrieval (Liau and Yao, 2001), used for query relaxation, restriction, and exemplar-based retrieval
- Thus far: little usage, examples are toy examples

Introduction

- Vagueness: statements involve concepts for which there is no exact definition, such as tall, close
- Statements are true to some degree which is taken from a *truth space*, which is usually $[0, 1]$
  - Hotel Verdi is close to the train station to degree 0.83
  - Find top-k cheapest hotels close to the train station: $q(h) \leftarrow hasLocation(h, hl) \wedge hasLocation(train, cl) \wedge close(hl, cl) \wedge cheap(h)$
  - What is the interpretation of $close(verdi, train) \wedge cheap(200)$?
  - Interpretation: a function I mapping atoms into $[0, 1]$, i.e. $I(A) \in [0, 1]$
  - if $I(close(verdi, train)) = 0.83$ and $I(cheap(200)) = 0.2$, then what is the result of $0.83 \wedge 0.2$?
- More generally, what is the result of $n \wedge m$, for $n, m \in [0, 1]$?

Fuzzy logic (basics)

- Formulae: First-Order Logic formulae, terms are either variables or constants
- many-valued formulae have the form $\phi \geq l$ or $\phi \leq u$ where $l, u \in [0, 1]$ (degree of truth is *at least* l and *at most* u, resp.)
- Formulae have a degree of truth in truth space $[0, 1]$
- Interpretation is a mapping $I : Atoms \rightarrow [0, 1]$, which are extended to formulae as follows (subsection):

$$\mathcal{I}(\neg\phi) = \mathcal{I}(\phi) \rightarrow 0 \quad (1)$$
$$\mathcal{I}(\exists x \phi) = \sup_{c \in \Delta^{\mathcal{I}}} \mathcal{I}_x^c(\phi) \quad (2)$$
$$\mathcal{I}(\forall x \phi) = \inf_{c \in \Delta^{\mathcal{I}}} \mathcal{I}_x^c(\phi) \quad (3)$$
$$\mathcal{I}(\phi \wedge \psi) = \mathcal{I}(\phi) \otimes \mathcal{I}(\psi) \quad (4)$$
$$\mathcal{I}(\phi \vee \psi) = \mathcal{I}(\phi) \oplus \mathcal{I}(\psi) \quad (5)$$
$$\mathcal{I}(\phi \rightarrow \psi) = \mathcal{I}(\phi) \Rightarrow \mathcal{I}(\psi) \quad (6)$$
$$\mathcal{I}(\neg\phi) = \ominus\mathcal{I}(\phi) \quad (7)$$

cont'd

- where $\mathcal{I}_x^c$ is as $\mathcal{I}$ except that var x is mapped to individual c
- $\otimes$, $\oplus$, $\Rightarrow$, and $\ominus$ are *combination functions*: triangular norms (or t-norms), triangular co-norms (or s-norms), implication functions, and negation functions, respectively
- which extend the classical Boolean conjunction, disjunction, implication, and negation, respectively, to the many-valued case
- Degree of subsumption between two fuzzy sets A and B, denoted $A \sqsubseteq B$, is defined as $\inf_{x \in X} A(x) \Rightarrow B(x)$
  - If $A(x) \leq B(x)$ for all $x \in [0, 1]$ then $A \sqsubseteq B$ evaluates to 1
- $\mathcal{I} \models \phi \geq l$ (resp. $\mathcal{I} \models \phi \leq u$) iff $\mathcal{I}(\phi) \geq l$ (resp. $\mathcal{I}(\phi) \leq u$)

Fuzzy RDF and RDFS

- Fuzzy RDF
- Statement (triples) may have attached a degree in $[0, 1]$: for $n \in [0, 1]$
  - $\langle (subject, predicate, object), n \rangle$
  - Meaning: the degree of truth of the statement is at least n
  - E.g.: $\langle (o1, isAbout, snoopy), 0.8 \rangle$
- Inferences, e.g.: $\frac{\langle (a, subClassOf, b), n \rangle, \langle (x, type, a), m \rangle}{\langle (x, type, b), n \wedge m \rangle}$
- Fuzzy RDFS adds extra constraints on interpretations
- see, e.g., 'A fuzzy semantics for semantic web languages' by Mazzieri and Dragoni, 2005

Fuzzy DLs can be classified according to:

- the description logic resp. ontology language that they generalize
- the allowed fuzzy constructs and the underlying fuzzy logics (Gödel, Lukasiewicz, Zadeh, ...)
- their reasoning services:
  - Consistency, Subsumption, Equivalence
  - Graded instantiation: Check if individual a is an instance of class C to degree at least n, i.e., $KB \models \langle a : C, n \rangle$
  - Best Truth Value Bound problem: determine tightest bound $n \in [0,1]$ of an axiom α, i.e. $glb(KB, \alpha) = sup\{n, | \ KB \models \langle \alpha \geq n \rangle\}$ (likewise for lub)
  - Best Satisfiability Bound problem: $glb(KB, C)$ determined by the max value of x s.t. $(\mathcal{R}, \mathcal{T}, \mathcal{A} \cup \{a : C \geq x\})$ (among all models, determine the max degree of truth that concept C may have over all individuals $x \in \Delta^{\mathcal{I}}$)
  - $glb(KB, C \sqsubseteq D)$ is the minimal value of x such that $KB = (\mathcal{R}, \mathcal{T}, \mathcal{A} \cup \{a : C \sqcap \neg D \geq 1 - x\})$ is satisfiable, where a is a new individual; Therefore, the greatest lower bound problem can be reduced to the minimal satisfiability problem of a fuzzy knowledge base

Fuzzy OWL

- Fuzzy $\mathcal{SHIF}(D)$, $\mathcal{SHOIN}(D)$, $\mathcal{SROIQ}(D)$, ...
- Additionally, we add
  - modifiers (e.g., very)
  - concrete fuzzy concepts (e.g., Young)
  - both additions have explicit membership functions

Concrete fuzzy concepts

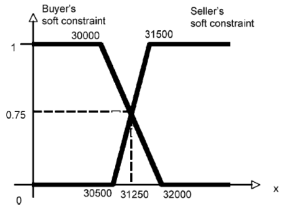
- Examples: Small, Young, High, Tall etc, with explicit membership function
- Use concrete domains to specify them:
  - $D = \langle \Delta_D, \Phi_D \rangle$, where Δ_D is an interpretation domain and Φ_D the set of concrete fuzzy domain predicates d with a predefined arity $n = 1, 2$ and *fixed* interpretation $d^D : \Delta_D^n \rightarrow [0,1]$
- For instance:
  - $\leq_{18}(x)$ over $\mathbb{N}$, evaluates to true if $x \leq 18$, false otherwise, or $cr(0, 18)$
  - Define $Minor \equiv Person \sqcap \exists Age.\leq_{18}$
  - Let $Young : Natural \rightarrow [0,1]$ be a fuzzy datatype predicate denoting the degree of youngness
  - Define $Young(x) = ls(x, 10, 30)$, where ls is the usual left shoulder function
  - Define $YoungPerson \equiv Person \sqcap \exists Age.Young$
  - Then, the KB entails, e.g.:
    $KB \models Minor \sqsubseteq YoungPerson \geq 0.6$,
    $YoungPerson \sqsubseteq Minor \geq 0.4$

Modifiers

- Very, moreOrLess, slightly, etc.
- Apply to fuzzy sets to change their membership function
  - fuzzy modifier m represents a function $f_m : [0,1] \rightarrow [0,1]$, with $\mathbf{M}$ an alphabet for fuzzy modifiers and $m \in \mathbf{M}$
  - then, if C is a concept in, say, fuzzy $\mathcal{SHOIN}$, then so is $m(C)$
  - Modifiers are definable as linear in-equations over $\mathbb{Q}$, $\mathbb{Z}$ (e.g., linear hedges), for instance, linear hedges[4], $lm(x; a, b)$, e.g. $very = lm(x; 0.7, 0.49)$
- Example:
  - $f_{very}(x) = x^2$
  - $f_{slightly}(x) = \sqrt{x}$
  - $SportsCar \equiv Car \sqcap \exists speed.very(High)$, where $very$ is the fuzzy modifier and $High$ a fuzzy datatype over the domain of speed (in km/h) and may be defined as, say, $High(x) = rs(80, 250)$

[4] they modify the shape of a fuzzy set in predictable ways; e.g., by pushing all values less than one towards zero, thereby shrinking the fuzzy part of the set closer to the area that is completely in the set

- The car seller and buyer revisited
- More fine grained approach: prices as vague constraints
  - Seller would sell above 31500 euro, but can go down to 30500
  - Buyer prefers to spend less than 30000, but can go up to 32000
    - $AudiTT \sqsubseteq SportsCar \sqcap \exists hasPrice.rs(30500, 31500)$
    - $Request \sqsubseteq SportsCar \sqcap \exists hasPrice.ls(30000, 32000)$
    - with *rs* right-shoulder function and *ls* the left-shoulder function
  - Highest degree to which $C \equiv AudiTT \sqcap Request$ is satisfiable is 0.75 (possibility that the Audi TT and the query matches is 0.75; the $glb(KB, C) = 0.75$)
  - the car may be sold at 31250 euro



Fuzzy OWL and reasoning

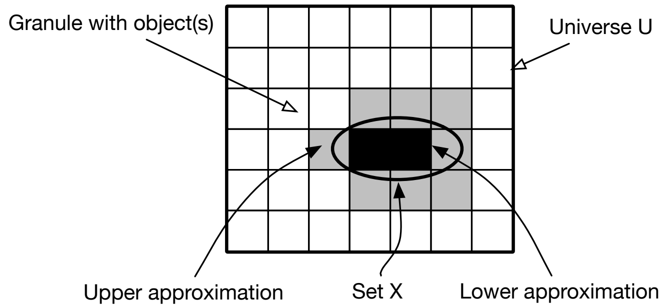
- Three principal approaches tested: Tableaux method, MILP based method, MIQP based method
- Implementation issues; Several options exists:
  - Try to map fuzzy DLs to classical DLs, but difficult to work with modifiers and concrete fuzzy concepts
  - Try to map fuzzy DLs to some fuzzy logic programming: A lot of work exists about mappings among classical DLs and LPs, but needs a theorem prover for fuzzy LPs
  - Build ad-hoc theorem prover for fuzzy DLs, using e.g., MILP
- A theorem prover for fuzzy $\mathcal{SHIF}$ + linear hedges + concrete fuzzy concepts + linear equational constraints + datatypes, under classical, Zadeh, Lukasiewicz and Product t-norm semantics has been implemented (`http://gaia.isti.cnr.it/~straccia`)
- FIRE: a fuzzy DL theorem prover for fuzzy $\mathcal{SHIN}$ under Zadeh semantics (`http://www.image.ece.ntua.gr/~nsimou/`)

Rough ontologies: Introduction

- Extension of rough sets to the knowledge representation layer; or: extension of crisp concepts in an ontology to incomplete data in the ABox
- Preliminary results with 'rough ontologies'
- Some results with fuzzy-rough and rough-fuzzy ontology languages
- No (end-user) tools and demonstration case studies in a subject domain yet: requires linking of ontology to sufficient data, i.e. **need for scalable semantic web technologies**

Rough sets

- Brief introduction of the Pawlak rough set model



Rough sets

- $I = (U, A)$ is called an *information system*, where U is a non-empty finite set of objects and A a finite non-empty set of attributes
- For every $a \in A$, function $a : U \mapsto V_a$ where v_a is the set of values that attribute a can have
- For any subset of attributes $P \subseteq A$, one can define the equivalence relation $\text{IND}(P)$ as

$$\text{IND}(P) = \{(x, y) \in U \times U \mid \forall a \in P, a(x) = a(y)\} \quad (8)$$

- $\text{IND}(P)$ generates a partition of U, which is denoted with $U/\text{IND}(P)$, or U/P for short.
- If $(x, y) \in \text{IND}(P)$, then x and y are indistinguishable with respect to the attributes in P, i.e, they are *p-indistinguishable*.

Rough sets (cont'd)

- From the objects in universe U, we want to represent set X such that $X \subseteq U$ using the attribute set P where $P \subseteq A$
- $[x]_P$ denotes the equivalence classes of the p-indistinguishability relation
- X may not be represented in a crisp way—the set may include and/or exclude objects which are indistinguishable on the basis of the attributes in P—but it can be approximated by using lower and upper approximation, respectively:

$$\underline{P}X = \{x \mid [x]_P \subseteq X\} \quad (9)$$
$$\overline{P}X = \{x \mid [x]_P \cap X \neq \emptyset\} \quad (10)$$

Rough sets (cont'd)

- The *lower approximation* is the set of objects that are *positively* classified as being members of set X, i.e., it is the union of all equivalence classes in $[x]_P$
- The *upper approximation* is the set of objects that are *possibly* in X
- Its complement, $U - \overline{P}X$, is the *negative region* with sets of objects that are definitely not in X (i.e., $\neg X$)
- with every rough set we associate two *crisp* sets, called *lower* and *upper approximation*, denoted as a tuple $X = \langle \underline{X}, \overline{X} \rangle$
- The difference between the lower and upper approximation, $B_P X = \overline{P}X - \underline{P}X$, is the *boundary region* of which its objects neither can be classified as to be member of X nor that they are not in X; if $B_P X = \emptyset$ then X is, in fact, a crisp set with respect to P and when $B_P X \neq \emptyset$ then X is rough w.r.t. P

Rough ontologies

- Several proposals, mainly DL+ rough extensions
- diverge in the formalisation what to include to represent roughness and thus also as to what rough concepts and rough ontologies actually are
- E is the symmetric, reflexive, transitive equivalence relation
- Let C_R be a (rough) concept in a DL language, then semantics for its lower and upper approximation are:

$$\underline{C} = \{x \mid \forall y : (x, y) \in E \rightarrow y \in C\} \quad (11)$$
$$\overline{C} = \{x \mid \exists y : (x, y) \in E \wedge y \in C\} \quad (12)$$

- Interpretation should map every approximate concept $C_R = \langle \underline{C}, \overline{C} \rangle$ to a pair over $\Delta^{\mathcal{I}}$, i.e., extending $\cdot^{\mathcal{I}}$ as follows:

$$C_R^{\mathcal{I}} = (\langle \underline{C}, \overline{C} \rangle)^{\mathcal{I}} = \langle (\underline{C})^{\mathcal{I}}, (\overline{C})^{\mathcal{I}} \rangle \quad (13)$$

- Interesting property: $C \sqsubseteq D \Rightarrow \langle \underline{C}, \overline{C} \rangle \sqsubseteq \langle \underline{D}, \overline{D} \rangle$

Reasoning services

- Alike the standard DL reasoning services:
  - approximate concept *satisability*, being the definitely satisability and possibly satisability (note that of C_R is possibly unsatisfiable, it is also definitely unsatisfiable)
  - approximate concepts rough *subsumption reasoning*
  - may be reduced to concept satisability reasoning problem in classical description logics (after transformation from RoughDL to standard DL)
- *Instance classification* of the objects into the approximations and their corresponding rough concepts

Applications

- None, i.e., no use case in a subject domain, except a few toy examples
- Potential: hypothesis testing, classification of patients, etc.
- Extending the theory: fuzzy-rough DL language (Bobillo and Straccia, 2009)
  - Extending also the reasoning algorithms
  - Add $\underline{C}$ and $\overline{C}$ represented as fuzzy DL concepts: $\overline{C}^i \mapsto \exists s_i.C$ and $\underline{C}_i \mapsto \forall s_i.C$, where s_i is a fuzzy similarity relation, add symmetry and reflexivity (in fuzzy $\mathcal{SHIF}(D)$, i.e. fuzzy OWL Lite)
  - Replace `(upper` s_i `C)` with `(some` s_i `C)` and `(lower` s_i `C)` with `(all` s_i `C)`, then add in the fuzzy RBox the reflexivity, symmetry and transitivity of R (in fuzzy $\mathcal{SROIQ}(D)$, i.e. fuzzy OWL 2 DL)
  - Then use the FUZZYDL and DELOREAN reasoners, respectively

A scenario

Suppose a person would like to “buy a sports car that costs at most about 22 000 euro and that has a power of around 150 HP”

- the buyer has to manually search for car selling web sites, e.g., using Google;
- select the most promising sites;
- browse through them, query them to see the cars that each site sells, and match the cars with the requirements;
- select the offers in each web site that match the requirements; and
- eventually merge all the best offers from each site and select the best ones.

A scenario

- A shopping agent automatizing the whole process once it receives the query q from the buyer:
  - The agent selects some resources S that it considers as relevant to q (*probabilistic*)
  - For the top-k selected sites, the agent reformulates q using the ontology of the specific car selling site (*probabilistic*)
  - q may contain many vague/fuzzy concepts (“around 150 HP”), so a car may *match q to a degree*. So, a resource returns a ranked list of cars, where the ranks depend on the degrees to which the cars match q. (*fuzzy*)
  - The agent integrates the ranked lists (using *probabilities*) and shows the top-n items to the buyer (or divided by *definite* and *possible* matches)
- To do this, there are bits and pieces for the languages and reasoners, but not everything *together*

Tools

- Probabilistic ontology tools:
  - Pronto: pellet + probabilistic http://pellet.owldl.com/pronto/
  - PR-OWL http://www.pr-owl.org/
  - Probabilisitc Ontology Alignment Tool http://gaia.isti.cnr.it/∼straccia/software/oMap/oMap.html
  - OMEN: A Probabilistic Ontology Mapping Tool
  - BayesOWL, OntoBayes
  - TOSS http://om.umiacs.umd.edu/ptoss.html
- Fuzzy ontology tools:
  - Fuzzy RDF http://gaia.isti.cnr.it/∼straccia/software/fuzzyRDF/fuzzyRDF.html
  - FUZZYDL http://gaia.isti.cnr.it/∼straccia/software/fuzzyDL/fuzzyDL.html
  - DELOREAN http://webdiis.unizar.es/∼fbobillo/delorean.php
  - FUZZY-KAZIMIR http://www.openclinical.org/prj_kasimir.html
  - FIRE http://www.image.ece.ntua.gr/∼nsimou/

Examples

- Probabilistic ontologies:
  - Star Trek ontology (experimental ontology to demonstrate PR-OWL) http://www.pr-owl.org/basics/ontostartrek.php
  - Astronomy to demonstrate TOSS http://om.umiacs.umd.edu/pparq.html
- Fuzzy ontologies:
  - Ontology Mediated Multimedia Information Retrieval System http://gaia.isti.cnr.it/∼straccia/software/DL-Media/DL-Media.html
  - Oncology with FUZZY-KAZIMIR http://www.oncolor.org/
  - FIRE with an medical imaging example

Summary