Fuzzy Semantic Web Languages and Beyond

Umberto Straccia

ISTI - CNR, Pisa, Italy
straccia@isti.cnr.it

www.umbertostraccia.it

About Vagueness

On the Existence of Vague Concepts

What are vague concepts and do they exists?

- What is this picture about?

- Vague concept: its extension is lacking in clarity
  - Aboutness of a picture or piece of text
  - Tall person
  - High temperature
  - Nice weather
  - Adventurous trip
- Vague concepts:
  - Are abundant in everyday speech and almost inevitable
  - Their meaning is often subjective and context dependent

What are vague objects and do they exists?

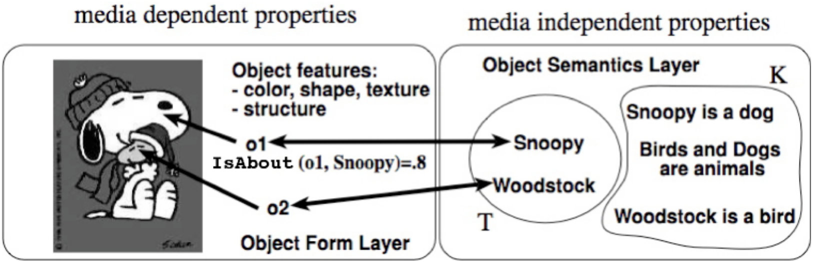
- Are there vague objects in the pictures?

- Vague object: its identity is lacking in clarity
  - Cloud
  - Dunes
  - Sun
- Vague objects:
  - Are not identical to anything, except to themselves (reflexivity)
  - Are characterised by a vague identity relation (e.g. a similarity relation)

Vague Statements

- A statement is vague whenever it involves vague concepts or vague objects
- The truth of a vague statement is a matter of degree,
  - it is intrinsically difficult to establish whether the statement is entirely true or false
  - The weather temperature is 33 °C. Is it hot?

Sources of Vagueness: Multimedia information retrieval



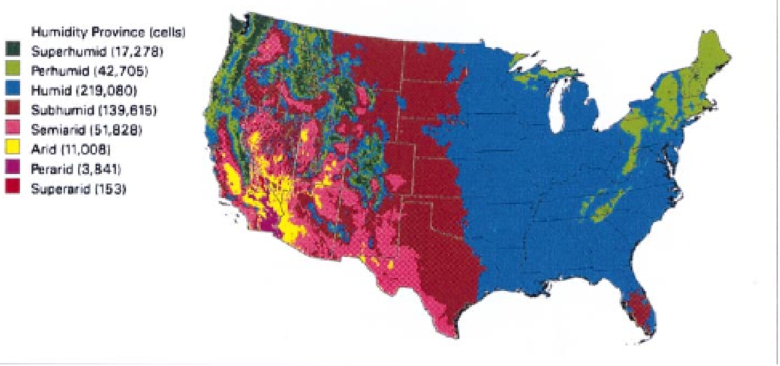
IsAbout		
ImageRegion	*Object ID*	*degree*
*o*1	*snoopy*	0.8
*o*2	*woodstock*	0.7
⋮	⋮	

"Find top-*k* image regions about animals"

$$Query(x) \leftarrow ImageRegion(x) \wedge isAbout(x, y) \wedge Animal(y)$$

Sources of Vagueness: Lifezone mapping

- To which degree do certain areas have a specific bioclima



Holdridge life zones of USA

Sources of Vagueness: ARPAT, Air quality in the province of Lucca

Sintesi dei dati rilevati dalle ore 0 alle ore 24 del giorno domenica 14/02/2010

Stazione		Tipo stazione	SO_2 µg/m^3 (media su 24h)	NO_2 µg/m^3 (max oraria)	CO mg/m^3 (max oraria)	O_3 µg/m^3 (max oraria)	PM_{10} µg/m^3 (media su 24h)	Giudizio di qualità dell'aria
Lucca	P.za San Micheletto (RETE REGIONALE **)	urbana - traffico	1	75	---	---	56	Scadente
Lucca	V.le Carducci	urbana - traffico	2	---	2	---	75	Pessima
Lucca	Carignano (RETE REGIONALE **)	rurale - fondo	---	---	---	87 (h.18*)	---	Buona
Viareggio	Largo Risorgimento	urbana - traffico	---	---	1,7	---	n.d.	Buona
Viareggio	Via Maroncelli (RETE REGIONALE **)	urbana - fondo	1	121	---	60 (h.17*)	45	Accettabile
Capannori	V. di Piaggia (RETE REGIONALE **)	urbana - fondo	---	79	2	---	59	Scadente
Porcari	V. Carrara (RETE REGIONALE **)	periferica - fondo	2	72	---	82 (h.16*)	63	Scadente

Giudizio di qualità	SO_2 µg/m^3 (media su 24h)	NO_2 µg/m^3 (max oraria)	CO mg/m^3 (max oraria)	O_3 µg/m^3 (max oraria)	PM_{10} µg/m^3 (media su 24h)
Buona	0-50	0-50	0-2,5	0-120	0-25
Accettabile	51-125	51-200	2,6-15	121-180	26-50
Scadente	126-250	201-400	15,1-30	181-240	51-74
Pessima	>250	>400	>30	>240	>74

http://www.arpat.toscana.it/

TripAdvisor: Hotel User Judgments

Uncertainty vs Vagueness: a clarification

- Initial difficulty:
  - Understand the conceptual differences between uncertainty and vagueness
- Main problem:
  - Interpreting a degree as a measure of uncertainty rather than as a measure of vagueness

Uncertain Statements

- A statement is true or false in any world/interpretation
  - We are “uncertain” about which world to consider as the actual one
  - We may have e.g. a probability/possibility distribution over possible worlds
- E.g., of uncertain statement: “it will rain tomorrow”
  - We cannot exactly establish whether it will rain tomorrow or not, due to our incomplete knowledge about our world
  - But, we may estimate to which degree this is e.g. probable/possible

Vague Statements

- A statement is vague if it involves vague concepts
- A statement is true to some degree, which is taken from a truth space (usually $[0, 1]$)
- E.g. of vague statement: “heavy rain”
  - is graded and the degree depends on the amount of rain is falling

In weather forecasts one may find:

Rain. Falling drops of water larger than 0.5 mm in diameter. "Rain" usually implies that the rain will fall steadily over a period of time;

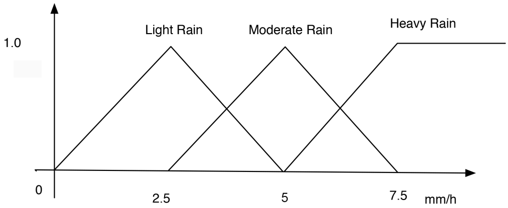
Light rain. Rain falls at the rate of 2.6 mm or less an hour;

Moderate rain. Rain falls at the rate of 2.7 mm to 7.6 mm an hour;

Heavy rain. Rain falls at the rate of 7.7 mm an hour or more.

- Quite harsh distinction:
$$\begin{array}{lcl} R = 7.7mm/h & \rightarrow & \text{heavy rain} \\ R = 7.6mm/h & \rightarrow & \text{moderate rain} \end{array}$$
- Unsatisfactory:
  - the more rain is falling, the more the sentence "heavy rain" is true
  - vice-versa, the less rain is falling the more the sentence "heavy rain" is false

- I.e., the sentence “heavy rain” is intrinsically graded
- More fine grained approach:
  - Define the various types of rains as



  - Light rain, moderate rain and heavy rain are vague concepts

- Are there sentences combining the two orthogonal concepts of uncertainty and vagueness?
- Yes, and we use them daily !
  - E.g. “*There will be heavy rain tomorrow*."
- This type of sentences are called uncertain vague sentences
- Essentially, there is
  - uncertainty about the world we will have tomorrow
  - vagueness about the various types of rain

From Fuzzy Sets to Mathematical Fuzzy Logic

Fuzzy Sets Basics

From Crisp Sets to Fuzzy Sets.

- Let X be a universal set of objects
- The power set, denoted 2^A, of a set $A \subset X$, is the set of subsets of A, i.e.,

$$2^A = \{B \mid B \subseteq A\}$$

- Often sets are defined as

$$A = \{x \mid P(x)\}$$

  - $P(x)$ is a statement "x has property P"
  - $P(x)$ is either true or false for any $x \in X$

- Examples of universe X and subsets $A, B \in 2^X$ may be

$$\begin{aligned} X &= \{x \mid x \text{ is a day}\} \\ A &= \{x \mid x \text{ is a rainy day}\} \\ B &= \{x \mid x \text{ is a day with precipitation rate } R \geq 7.5mm/h\} \end{aligned}$$

- In the above case: $B \subseteq A \subseteq X$
- The membership function of a set $A \subseteq X$:

$$\chi_A \colon X \to \{0, 1\}$$

where $\chi_A(x) = 1$ iff $x \in A$

- Complement of a set A, i.e. $\bar{A} = X \setminus A$: $\forall x \in X$:

$$\chi_{\bar{A}}(x) = 1 - \chi_A(x)$$

- Intersection and union: $\forall x \in X$

$$\begin{aligned} \chi_{A \cap B}(x) &= \min(\chi_A(x), \chi_B(x)) \\ \chi_{A \cup B}(x) &= \max(\chi_A(x), \chi_B(x)) \end{aligned}$$

- Fuzzy set A: $\chi_A\colon X \to [0,1]$, or simply

$$A\colon X \to [0,1]$$

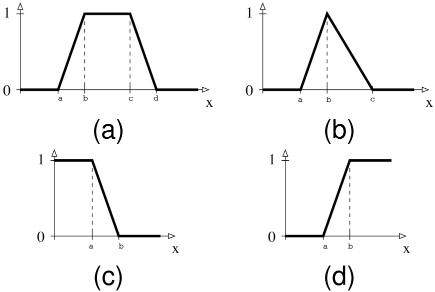
- Example: the fuzzy set

$$C = \{x \mid x \text{ is a day with heavy precipitation rate } R\}$$

is defined via the membership function

$$\chi_C(x) = \begin{cases} 1 & \text{if } R \geq 7.5 \\ (x-5)/2.5 & \text{if } R \in [5, 7.5) \\ 0 & \text{otherwise} \end{cases}$$

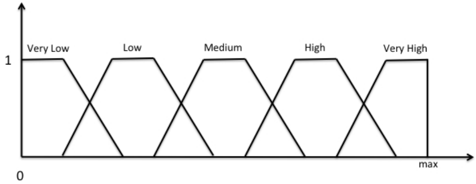
- Fuzzy membership functions may depend on the context and may be subjective
- Shape may be quite different
- Usually, it is sufficient to consider functions



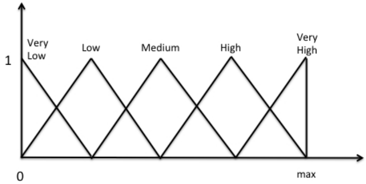
(a) Trapezoidal $trz(a, b, c, d)$; (b) Triangular $tri(a, b, c)$; (c) left-shoulder $ls(a, b)$; (d) right-shoulder $rs(a, b)$

Fuzzy Sets Construction

- Simple and typically satisfactory method (numerical domain):
  - uniform partitioning into 5 fuzzy sets

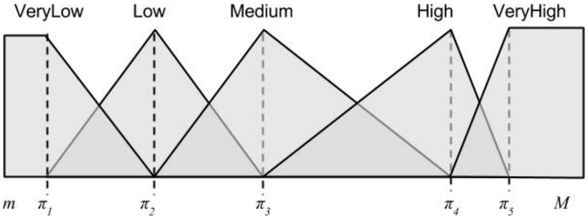


Fuzzy sets construction using trapezoidal functions



Fuzzy sets construction using triangular functions

- Another popular method is based on clustering
- Use Fuzzy C-Means to cluster data into 5 clusters
  - Fuzzy C-Means extends K-Means to accommodates graded membership
- From the clusters $c_1, \ldots, c_5$ take the centroids $\pi_1, \ldots, \pi_5$
- Build the fuzzy sets from the centroids



Fuzzy sets construction using clustering

Norm-Based Fuzzy Set Operations

- Standard fuzzy set operations are not the only ones
- Most notable ones are triangular norms
  - t-norm $\otimes$ for set intersection
  - t-conorm $\oplus$ (also called s-norm) for set union
  - negation $\ominus$ for set complementation
  - implication $\Rightarrow$ for set inclusion
- These functions satisfy some properties that one expects to hold

Łukasiewicz, Gödel, Product logic and Standard Fuzzy logic

- One distinguishes three different sets of fuzzy set operations (called fuzzy logics)
  - Łukasiewicz, Gödel, and Product logic
  - Standard Fuzzy Logic (SFL) is a sublogic of Łukasiewicz
    - $\min(a, b) = a \otimes_l (a \Rightarrow_l b)$, $\max(a, b) = 1 - \min(1 - a, 1 - b)$

	Łukasiewicz Logic	Gödel Logic	Product Logic	SFL
$a \otimes b$	$\max(a + b - 1, 0)$	$\min(a, b)$	$a \cdot b$	$\min(a, b)$
$a \oplus b$	$\min(a + b, 1)$	$\max(a, b)$	$a + b - a \cdot b$	$\max(a, b)$
$a \Rightarrow b$	$\min(1 - a + b, 1)$	$\begin{cases} 1 & \text{if } a \leq b \\ b & \text{otherwise} \end{cases}$	$\min(1, b/a)$	$\max(1 - a, b)$
$\ominus a$	$1 - a$	$\begin{cases} 1 & \text{if } a = 0 \\ 0 & \text{otherwise} \end{cases}$	$\begin{cases} 1 & \text{if } a = 0 \\ 0 & \text{otherwise} \end{cases}$	$1 - a$

- Mostert–Shields theorem: any continuous t-norm can be obtained as an ordinal sum of Ł, G and P.

Mathematical Fuzzy Logics Basics

- OWL 2 is grounded on Mathematical Logic
- Fuzzy OWL 2 is grounded on Mathematical Fuzzy Logic
- A statement is graded
- Truth space: set of truth values L
- Given a statement ϕ
  - Fuzzy Interpretation: a function $\mathcal{I}$ mapping ϕ into L, i.e.

$$\mathcal{I}(\varphi) \in L$$

  - Usually

$$\begin{array}{rcl} L & = & [0, 1] \\ L_n & = & \{0, \frac{1}{n}, \ldots, \frac{n-2}{n-1}, \ldots, 1\} \quad (n \geq 1) \end{array}$$

- Fuzzy statement: for formula ϕ and $r \in [0, 1]$

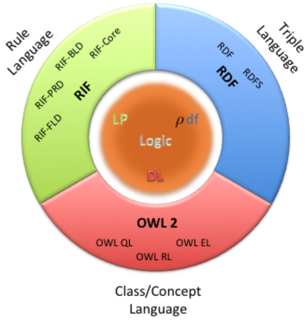
$$\langle \phi, r \rangle$$

The degree of truth of ϕ is equal or greater than r

Fuzzy Semantic Web Languages and Beyond

The Semantic Web Family of Languages

- Wide variety of languages
  - RDFS: *Triple language*, *-Resource Description Framework*
    - The logical counterpart is ρdf
  - RIF: *Rule language*, *-Rule Interchange Format*,
    - Relate to the *Logic Programming* (LP) paradigm
  - OWL 2: *Conceptual language*, *-Ontology Web Language*
    - Relate to Description Logics (DLs)



RDFS

- RDFS: the triple language

$$\langle \mathit{subject}, \mathit{predicate}, \mathit{object} \rangle$$

e.g. $\langle \mathit{umberto}, \mathit{born}, \mathit{zurich} \rangle$

- OWL 2 family: an object oriented language

```
class   PERSON partial
        restriction (hasName someValuesFrom String)
        restriction (hasBirthPlace someValuesFrom GEOPLACE)
        ...
```

OWL 2 Profiles

OWL 2 EL

- Useful for large size of properties and/or classes
- The EL acronym refers to the $\mathcal{EL}$ family of DLs

OWL 2 QL

- Useful for very large volumes of instance data
- Conjunctive query answering via query rewriting and SQL
- OWL 2 QL relates to the DL family *DL-Lite*

OWL 2 RL

- Useful for scalable reasoning without sacrificing too much expressive power
- OWL 2 RL maps to Datalog

RIF/RuleML

- **RIF/RuleML** family: the rule language

  `Forall` ?Buyer ?Item ?Seller
  buy(?Buyer ?Item ?Seller) :- sell(?Seller ?Item ?Buyer)

Important point: RDFS, OWL 2 and RIF/RuleML are logical languages

- RDFS: logic with intensional semantics
- OWL 2: relates to the Description Logics family
- RIF/RuleML: relates to the Logic Programming paradigm (e.g., Datalog, Datalog$^{\pm}$)
- OWL 2 and RIF/RuleML have extensional semantics

The case of RDF

Crisp RDFS Syntax

- RDFS triple (or RDFS atom):

$$(s, p, o)$$

  - s is the subject
  - p is the predicate
  - o is the object
- Example:

$$(\mathit{airplane}, \mathit{has}, \mathit{enginefault})$$

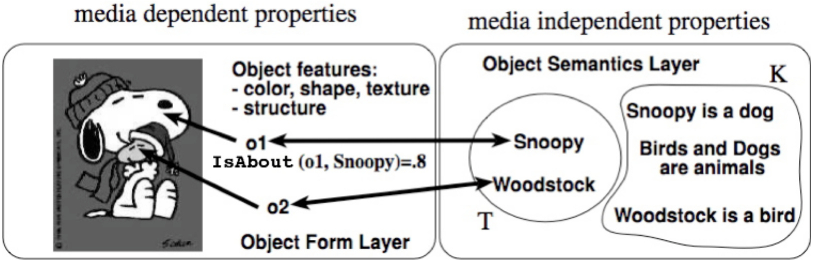
ρdf (restricted RDFS)

- ρdf (read rho-df, the ρ from restricted rdfs)
- ρdf is defined as the following subset of the RDFS vocabulary:

$$\rho\text{df} = \{\text{sp}, \text{sc}, \text{type}, \text{dom}, \text{range}\}$$

- (p, sp, q)
  - property p is a *sub property* of property q
- (c, sc, d)
  - class c is a *sub class* of class d
- (a, type, b)
  - a is of *type* b
- (p, dom, c)
  - *domain* of property p is c
- (p, range, c)
  - *range* of property p is c

Example



$$G = \left\{ \begin{array}{ll} (o1, IsAbout, snoopy) & (o2, IsAbout, woodstock) \\ (snoopy, \text{type}, dog) & (woodstock, \text{type}, bird) \\ (dog, \text{sc}, animal) & (bird, \text{sc}, animal) \end{array} \right\}$$

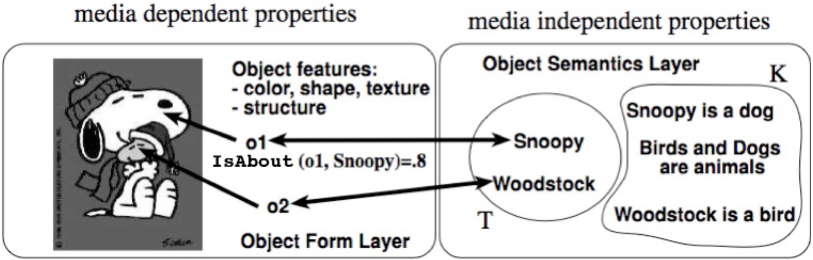
RDFS Query Answering

- Conjunctive query: is a Datalog-like rule of the form

$$q(\mathbf{x}) \leftarrow \exists \mathbf{y}.\tau_1, \ldots, \tau_n$$

where $\tau_1, \ldots, \tau_n$ are triples in which variables in $\mathbf{x}$ and $\mathbf{y}$ may occur (we may omit $\exists \mathbf{y}$)

Example



$$G = \left\{ \begin{array}{ll} (o1, \mathit{IsAbout}, \mathit{snoopy}) & (o2, \mathit{IsAbout}, \mathit{woodstock}) \\ (\mathit{snoopy}, \mathrm{type}, \mathit{dog}) & (\mathit{woodstock}, \mathrm{type}, \mathit{bird}) \\ (\mathit{dog}, \mathrm{sc}, \mathit{animal}) & (\mathit{bird}, \mathrm{sc}, \mathit{animal}) \end{array} \right\}$$

Query:

$$q(x) \quad \leftarrow \quad (x, \mathit{IsAbout}, y), (y, \mathrm{type}, \mathit{Animal})$$

Then

$$\mathit{answer}(G, q) \quad = \quad \{o1, o2\}$$

Fuzzy RDFS

- Triples may have attached a degree n in L or L_n

$$\langle (subject, predicate, object), n \rangle$$

- Meaning: the degree of truth of the statement is at least n
- Example:

$$\langle (o1, IsAbout, snoopy), 0.8 \rangle$$

- How to represent fuzzy triples in RDFS?
  - Use reification method:

$$(s1, hasObj, o1), (s1, hasRel, IsAbout), (s1, hasObj, snoopy), (s1, hasDeg 0.8)$$

- Unfortunately, RDFS is lacking the "annotation property" of triples

Fuzzy RDFS Query Answering

- Conjunctive query: extends a crisp RDF query and is of the form

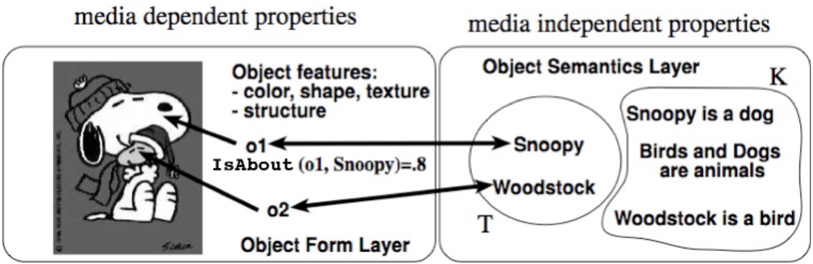
$$\langle q(\mathbf{x}), s\rangle \quad \leftarrow \quad \exists \mathbf{y}.\langle \tau_1, s_1\rangle, \ldots, \langle \tau_n, s_n\rangle, \\ s = f(s_1, \ldots, s_n, p_1(\mathbf{z}_1), \ldots, p_h(\mathbf{z}_h))$$

  where
  - τ_i triples involving literals and variables in $\mathbf{x}, \mathbf{y}$
  - $\mathbf{z_i}$ are tuples of literals or variables in $\mathbf{x}$ or $\mathbf{y}$
  - $p_j(\mathbf{t}) \in [0, 1]$
  - f is a *scoring* function $f\colon ([0, 1])^{n+h} \to [0, 1]$
- Example:

$$\langle q(x), s\rangle \leftarrow \langle (x, \text{type}, \text{SportCar}), s_1\rangle, (x, \text{hasPrice}, y), s = s_1 \cdot \text{cheap}(y)$$

  where e.g. $\text{cheap}(y) = \text{ls}(0, 10000, 12000)(y)$, has intended meaning to “retrieve all cheap sports car"

Example



$$G = \left\{ \begin{array}{ll} \langle (o1, IsAbout, snoopy), 0.8 \rangle & \langle (o2, IsAbout, woodstock), 0.9 \rangle \\ (snoopy, \mathrm{type}, dog) & (woodstock, \mathrm{type}, bird) \\ \langle (Dog, \mathrm{sc}, SmallAnimal), 0.4 \rangle & \langle (Bird, \mathrm{sc}, SmallAnimal), 0.7 \rangle \\ (SmallAnimal, \mathrm{sc}, Animal) & \end{array} \right\}$$

Consider the query

$$\langle q(x), s \rangle \quad \leftarrow \quad \langle (x, IsAbout, y), s_1 \rangle, \langle (y, \mathrm{type}, Animal), s_2 \rangle, s = s_1 \cdot s_2$$

Then

$$ans(G, q) \quad = \quad \{ \langle o1, 0.32 \rangle, \langle o2, 0.63 \rangle \}$$

Annotation domains & RDFS

- Generalisation of fuzzy RDFS
  - a triple is annotated with a value taken from a so-called annotation domain, rather than with a value in [0,1]
  - allows to deal with several domains (such as, fuzzy, temporal, provenance) and their combination, in a uniform way
- Fuzzyness
  - $\langle (\mathit{HolidayInnHotel}, \mathit{closeTo}, \mathit{IEA17Venue}), 0.7 \rangle$
  - true to some degree
- Time
  - $\langle (\mathit{umberto}, \mathit{workedFor}, \mathit{IEI}), [1992, 2001] \rangle$
  - true during 1992–2001
- Provenance
  - $\langle (\mathit{umberto}, \mathit{knows}, \mathit{salem}), \text{http://www.straccia.info/foaf.rdf} \rangle$
  - true in `http://www.straccia.info/foaf.rdf`
- Multiple Domains:

$$\langle (\mathit{CountryXYZ}, \text{type}, \mathit{Dangerous}), \langle [1975, 1983], 0.8, 0.6 \rangle \rangle$$

$\mathit{Time} \times \mathit{Fuzzy} \times \mathit{Trust}$

- Annotation Domain: idempotent, commutative semi-ring

$$D = \langle L, \oplus, \otimes, \bot, \top \rangle$$

  where $\oplus$ is $\top$-annihilating, i.e.
  1. $\oplus$ is idempotent, commutative, associative;
  2. $\otimes$ is commutative and associative;
  3. $\bot \oplus \lambda = \lambda$, $\top \otimes \lambda = \lambda$, $\bot \otimes \lambda = \bot$, and $\top \oplus \lambda = \top$;
  4. $\otimes$ is distributive over $\oplus$,
     i.e. $\lambda_1 \otimes (\lambda_2 \oplus \lambda_3) = (\lambda_1 \otimes \lambda_2) \oplus (\lambda_1 \otimes \lambda_3)$;
- Induced partial order:

$$\lambda_1 \preceq \lambda_2 \iff \lambda_1 \oplus \lambda_2 = \lambda_2$$

- Annotated triple: for $\lambda \in L$

$$\langle (s, p, o), \lambda \rangle$$

The case of Description Logics

Description Logics (DLs)

- Concept/Class: are unary predicates
- Role or attribute: binary predicates
- Taxonomy: Concept and role hierarchies can be expressed
- Individual: constants
- Operators: to build complex classes out from class names

- Basic ingredients:
  - a:C, meaning that individual a is an instance of concept/class C

$$\text{a:Person} \sqcap \exists\text{hasChild.Femal}$$

  - (a, b):R, meaning that the pair of individuals $\langle a, b\rangle$ is an instance of the property/role R

$$(\text{tom}, \text{mary})\text{:hasChild}$$

  - $C \sqsubseteq D$, meaning that the class C is a subclass of class D

$$\text{Father} \sqsubseteq \mathit{Male} \sqcap \exists\text{hasChild.Person}$$

The DL Family

- A given DL is defined by set of concept and role forming operators
- Basic language: $\mathcal{ALC}$ ($\mathcal{A}$ttributive $\mathcal{L}$anguage with $\mathcal{C}$omplement)

Syntax				Semantics	Example
C, D	$\rightarrow$	$\top$	$\mid$	$\top(x)$	
		$\bot$	$\mid$	$\bot(x)$	
		A	$\mid$	$A(x)$	*Human*
		$C \sqcap D$	$\mid$	$C(x) \wedge D(x)$	$Human \sqcap Male$
		$C \sqcup D$	$\mid$	$C(x) \vee D(x)$	$Nice \sqcup Rich$
		$\neg C$	$\mid$	$\neg C(x)$	$\neg Meat$
		$\exists R.C$	$\mid$	$\exists y.R(x,y) \wedge C(y)$	$\exists has\_child.Blond$
		$\forall R.C$		$\forall y.R(x,y) \Rightarrow C(y)$	$\forall has\_child.Human$
$C \sqsubseteq D$				$\forall x.C(x) \Rightarrow D(x)$	$Happy\_Father \sqsubseteq Man \sqcap \exists has\_child.Female$
$a{:}C$				$C(a)$	$John{:}Happy\_Father$

Note on DL Naming

$\mathcal{AL}$: $C, D \longrightarrow \top \mid \bot \mid A \mid C \sqcap D \mid \neg A \mid \exists R.\top \mid \forall R.C$

$\mathcal{C}$: Concept negation, $\neg C$. Thus, $\mathcal{ALC} = \mathcal{AL} + \mathcal{C}$

$\mathcal{S}$: Used for $\mathcal{ALC}$ with transitive roles $\mathcal{R}_+$

$\mathcal{U}$: Concept disjunction, $C_1 \sqcup C_2$

$\mathcal{E}$: Existential quantification, $\exists R.C$

$\mathcal{H}$: Role inclusion axioms, $R_1 \sqsubseteq R_2$, e.g. $is\_component\_of \sqsubseteq is\_part\_of$

$\mathcal{N}$: Number restrictions, $(\geq n\ R)$ and $(\leq n\ R)$, e.g. $(\geq 3\ has\_Child)$ (has at least 3 children)

$\mathcal{Q}$: Qualified number restrictions, $(\geq n\ R.C)$ and $(\leq n\ R.C)$, e.g. $(\leq 2\ has\_Child.Adult)$ (has at most 2 adult children)

$\mathcal{O}$: Nominals (singleton class), $\{a\}$, e.g. $\exists has\_child.\{mary\}$. **Note**: $a{:}C$ equiv to $\{a\} \sqsubseteq C$ and $(a, b){:}R$ equiv to $\{a\} \sqsubseteq \exists R.\{b\}$

$\mathcal{I}$: Inverse role, R^-, e.g. $isPartOf = hasPart^-$

$\mathcal{F}$: Functional role, f, e.g. $functional(hasAge)$

$\mathcal{R}_+$: transitive role, e.g. $transitive(isPartOf)$

For instance,

$\mathcal{SHIF}$	$=$	$\mathcal{S}+\mathcal{H}+\mathcal{I}+\mathcal{F} = \mathcal{ALCR_+HIF}$	OWL-Lite
$\mathcal{SHOIN}$	$=$	$\mathcal{S}+\mathcal{H}+\mathcal{O}+\mathcal{I}+\mathcal{N} = \mathcal{ALCR_+HOIN}$	OWL-DL
$\mathcal{SROIQ}$	$=$	$\mathcal{S}+\mathcal{R}+\mathcal{O}+\mathcal{I}+\mathcal{Q} = \mathcal{ALCR_+ROIN}$	OWL 2

Semantics of Additional Constructs

$\mathcal{H}$: Role inclusion axioms, $\mathcal{I} \models R_1 \sqsubseteq R_2$ iff ${R_1}^{\mathcal{I}} \subseteq {R_1}^{\mathcal{I}}$

$\mathcal{N}$: Number restrictions,
$(\geq n\ R)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : |\{y \mid \langle x, y \rangle \in R^{\mathcal{I}}\}| \geq n\}$,
$(\leq n\ R)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : |\{y \mid \langle x, y \rangle \in R^{\mathcal{I}}\}| \leq n\}$

$\mathcal{Q}$: Qualified number restrictions,
$(\geq n\ R.C)^{\mathcal{I}} = \{x \in |\{y \mid \langle x, y \rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}| \geq n\}$,
$(\leq n\ R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : |\{y \mid \langle x, y \rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}| \leq n\}$

$\mathcal{O}$: Nominals (singleton class), $\{a\}^{\mathcal{I}} = \{a^{\mathcal{I}}\}$

$\mathcal{I}$: Inverse role, $(R^{-})^{\mathcal{I}} = \{\langle x, y \rangle \mid \langle y, x \rangle \in R^{\mathcal{I}}\}$

$\mathcal{F}$: Functional role, $I \models fun(f)$ iff $\forall z \forall y \forall z$ if $\langle x, y \rangle \in f^{\mathcal{I}}$ and $\langle x, z \rangle \in f^{\mathcal{I}}$ the $y = z$

$\mathcal{R}_{+}$: transitive role,
$(R_{+})^{\mathcal{I}} = \{\langle x, y \rangle \mid \exists z$ such that $\langle x, z \rangle \in R^{\mathcal{I}} \wedge \langle z, y \rangle \in R^{\mathcal{I}}\}$

Basics on Concrete Domains

- Concrete domains: reals, integers, strings, ...

  $(tim, 14)$:*hasAge*
  $(sf, \text{``}SoftComputing\text{''})$:*hasAcronym*
  $(source1, \text{``}ComputerScience\text{''})$:*isAbout*
  $(service2, \text{``}InformationRetrievalTool''})$:*Matches*
  $Minor = Person \sqcap \exists hasAge. \leq_{18}$

- Notation: (D). E.g., $\mathcal{ALC}(D)$ is $\mathcal{ALC}$ + concrete domains

DL Knowledge Base

- A DL Knowledge Base is a pair $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, where
  - $\mathcal{T}$ is a TBox
    - containing general inclusion axioms of the form $C \sqsubseteq D$,
    - concept definitions of the form $A = C$
    - primitive concept definitions of the form $A \sqsubseteq C$
    - role inclusions of the form $R \sqsubseteq P$
    - role equivalence of the form $R = P$
  - $\mathcal{A}$ is a ABox
    - containing assertions of the form $a{:}C$
    - containing assertions of the form $(a, b){:}R$

Fuzzy DLs Basics

For a degree n in L or L_n

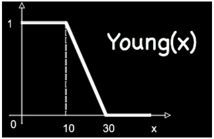
- $\langle a{:}C, n\rangle$ states that a is an instance of concept/class C with degree at least n
- $\langle (a, b){:}R, n\rangle$ states that $\langle a, b\rangle$ is an instance of relation R with degree at least n
- $\langle C_1 \sqsubseteq C_2, n\rangle$ states a vague subsumption relationship
  - The FOL statement $\forall x.C_1(x) \rightarrow C_2(x)$ is true to degree at least n

Towards Fuzzy OWL 2 and its Profiles

- Fuzzy OWL 2 added value:
  - fuzzy concrete domains (e.g., *young*)
  - modifiers (e.g., *very* young)
  - other extensions:
    - aggregation functions: weighted sum, OWA, fuzzy integrals
    - fuzzy rough sets
    - fuzzy spatial relations
    - fuzzy numbers, ...

Fuzzy Concrete Domains

- E.g., *Small*, *Young*, *High*, *etc*. with explicit membership function
- Representation of Young Person:



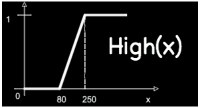
$$
\begin{aligned}
Minor &= Person \sqcap \exists hasAge. \leq_{18} \\
YoungPerson &= Person \sqcap \exists hasAge.ls(10, 30)
\end{aligned}
$$

- Representation of Heavy Rain:

$$HeavyRain = Rain \sqcap \exists hasPrecipitationRate.rs(5, 7.5)$$

Fuzzy Modifiers

- *Very*, *moreOrLess*, *slightly*, etc.
- Representation of Sport Car



$$SportsCar = Car \sqcap \exists speed.very(rs(80, 250))$$

- Representation of Very Heavy Rain

$$VeryHeavyRain = Rain \sqcap \exists hasPrecipitationRate.very(rs(5, 7.5)) \ .$$

Aggregation Operators

- Aggregation operators: aggregate concepts, using functions such as the mean, median, weighted sum operators, etc.
- Allows to express the concept

$$0.3 \cdot \mathit{ExpensiveHotel} + 0.7 \cdot \mathit{LuxuriousHotel} \sqsubseteq \mathit{GoodHotel}$$

  - a good hotel is the weighted sum of being an expensive and luxurious hotel
- Aggregated concepts are popular in robotics:
  - to recognise complex objects from atomic ones

Fuzzy DLs Query Answering

- Conjunctive query: similar to fuzzy RDFS CQs:

$$\begin{array}{lll}\langle q(\mathbf{x}), s\rangle & \leftarrow & \exists \mathbf{y}.\langle \tau_1, s_1\rangle, \ldots, \langle \tau_n, s_n\rangle, \\ & & s = f(s_1, \ldots, s_n, p_1(\mathbf{z}_1), \ldots, p_h(\mathbf{z}_h))\end{array}$$

  where

  - $\tau_1, \ldots, \tau_n$ are expressions $A(z)$ or $R(z, z')$, where A is a concept name, R is a role name, z, z' are individuals or variables in **x** or **y**

- Example:

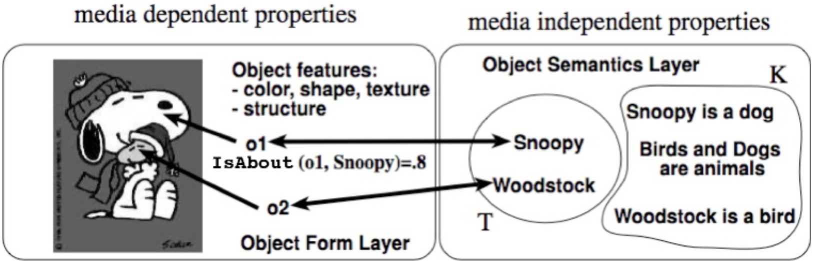
  $$\langle q(x), s\rangle \leftarrow \langle \mathrm{SportCar}(x), s_1\rangle, \mathrm{hasPrice}(x, y), s = s_1 \cdot \mathrm{cheap}(y)$$

  where e.g. $\mathrm{cheap}(y) = \mathrm{ls}(10000, 12000)(y)$, has intended meaning to retrieve all cheap sports car.

Some Applications

- (Multimedia) Information retrieval
- Recommendation systems
- Image interpretation
- Ambient intelligence
- Ontology merging
- Matchmaking
- Decision making
- Summarization
- Robotics perception
- Software design
- Machine learning

Example



$$G = \left\{ \begin{array}{ll} \langle (o1, snoopy){:}IsAbout, 0.8\rangle & \langle (o2, woodstock){:}IsAbout, 0.9\rangle \\ snoopy{:}Dog & woodstock{:}Bird \\ \langle Dog \sqsubseteq SmallAnimal, 0.4\rangle & \langle Bird \sqsubseteq SmallAnimal, 0.7\rangle \\ SmallAnimal \sqsubseteq Animal & \end{array} \right\}$$

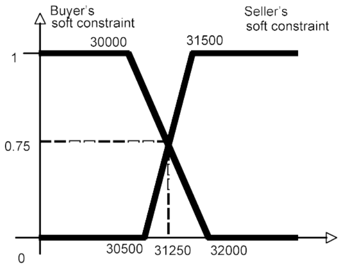
Consider the query

$$\langle q(x), s\rangle \quad \leftarrow \quad \langle IsAbout(x, y), s_1\rangle, \langle Animal(y), s_2\rangle, s = s_1 \cdot s_2$$

Then

$$ans(G, q) \quad = \quad \{\langle o1, 0.32\rangle, \langle o2, 0.63\rangle\}, \quad ans_1(G, q) = \{\langle o2, 0.63\rangle\}$$

Example (Simplified Matchmaking)



- A car seller sells an Audi TT for 31500 €, as from the catalog price.
- A buyer is looking for a sports-car, but wants to to pay not more than around 30000 €
- Classical sets: the problem relies on the crisp conditions on price

- More fine grained approach: to consider prices as fuzzy sets (as usual in negotiation)
  - Seller may consider optimal to sell above 31500 €, but can go down to 30500 €
  - The buyer prefers to spend less than 30000 €, but can go up to 32000 €

    $$\begin{array}{lcl} \mathit{AudiTT} & = & \mathit{SportsCar} \sqcap \exists \mathit{hasPrice}.\mathit{rs}(30500, 31500) \\ \mathit{Query} & = & \mathit{SportsCar} \sqcap \exists \mathit{hasPrice}.\mathit{ls}(30000, 32000) \end{array}$$

  - Highest degree to which the concept
    $C = \mathit{AudiTT} \sqcap \mathit{Query}$
    is satisfiable is 0.75 (the degree to which the Audi TT and the query matches is 0.75)

  - The car may be sold at 31250 €

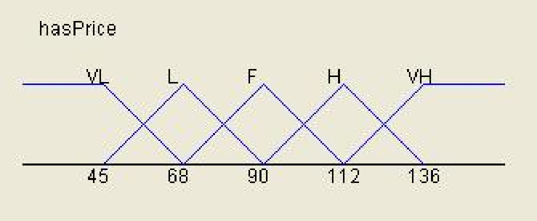
Example: Learning fuzzy GCIs from OWL data

- Learning of fuzzy GCIs from crisp OWL data
- Use Case: What are Good hotels, using TripAdvisor data?
  - Given
    - OWL 2 Ontology about meaningful city entities and their descriptions
    - TripAdvisor data about hotels and user judgments
  - We have learnt that in e.g., Pisa, Italy

$$\langle \exists hasAmenity.Babysitting \sqcap \exists hasPrice.fair \sqsubseteq Good\_Hotel, 0.782 \rangle$$

"A hotel having babysitting as amenity and a fair price is a good hotel (to degree 0.782)"

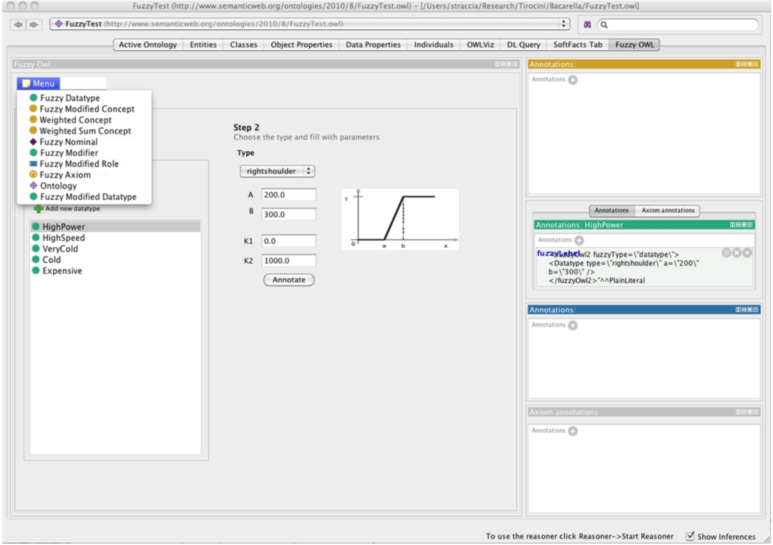
  - Real valued price attribute *hasPrice* has been automatically fuzzyfied



Representing Fuzzy OWL Ontologies in OWL

- OWL 2 is W3C standard, with classical logic semantics
  - Hence, cannot support natively Fuzzy Logic
- However, Fuzzy OWL 2, has been defined using OWL 2
  - Uses the axiom annotation feature of OWL 2
- Any Fuzzy OWL 2 ontology is a legal OWL 2 ontology

- A java parser for Fuzzy OWL 2 exists
- Protégé plug-in exists to encode Fuzzy OWL ontologies



Annotation domains & OWL

- For OWL 2, it it is like for RDFS, but annotation domain has to be a complete lattice
- Exception for OWL profiles OWL EL, OWL QL and OWL RL: annotation domains may be as for RDFS

The case of Logic Programs

LPs Basics (for ease, Datalog)

- Predicates are n-ary
- Terms are variables or constants
- Facts ground atoms
  For instance,
  $$has\_parent(mary, jo)$$
- Rules are of the form
  $$P(\mathbf{x}) \leftarrow \varphi(\mathbf{x}, \mathbf{y})$$
  where $\varphi(\mathbf{x}, \mathbf{y})$ is a formula built from atoms of the form $B(\mathbf{z})$
  For instance,
  $$has\_father(x, y) \quad \leftarrow \quad has\_parent(x, y), Male(y)$$

- Extensional database (EDB): set of facts
- Intentional database (IDB): set of rules
- Logic Program $\mathcal{P}$:
  - $\mathcal{P} = EDB \cup IDB$
  - No predicate symbol in *EDB* occurs in the head of a rule in *IDB*
    - The principle is that we do not allow that *IDB* may redefine the extension of predicates in *EDB*
- *EDB* is usually, stored into a relational database

LP Query Answering

- Query: is a rule of the form

$$q(\mathbf{x}) \leftarrow \varphi(\mathbf{x}, \mathbf{y})$$

- If $\mathcal{P} \models q(\mathbf{c})$ then $\mathbf{c}$ is called as usual an answer to q
- The answer set of q w.r.t. $\mathcal{P}$ is defined as

$$ans(\mathcal{P}, q) = \{\mathbf{c} \mid \mathcal{P} \models q(\mathbf{c})\}$$

Fuzzy LPs Basics

- Truth space is $[0,1]$ or $\{0, \frac{1}{n}, \ldots, \frac{n-2}{n-1}, \ldots, 1\}$ $(n \geq 1)$
- Generalized LP rules are of the form

$$\begin{array}{lll} \langle R(\mathbf{x}), s\rangle & \leftarrow & \exists \mathbf{y}.\langle R_1(\mathbf{z}_1), s_1\rangle, \ldots, \langle R_k(\mathbf{z}_k), s_k\rangle, \\ & & s = f(s_1, ..., s_k, p_1(\mathbf{z}'_1), \ldots, p_h(\mathbf{z}'_h)) \end{array}$$

  - Meaning of rules: "take the truth-values of all $R_i(\mathbf{z}_i), p_j(\mathbf{z}'_j)$, combine them using the truth combination function f, and assign the result to $R(\mathbf{x})$"
- Facts: ground expressions of the form $\langle R(\mathbf{c}), n\rangle$
  - Meaning of facts: "the degree of truth of $R(\mathbf{c})$ is at least n"
- Fuzzy LP: a set of facts (extensional database) and a set of rules (intentional database). No extensional relation may occur in the head of a rule

Example: Soft shopping agent

- User preferences:

$$\begin{aligned}\langle Pref_1(x,p), s\rangle &\leftarrow HasPrice(x,p), s = ls(10000, 14000)(p)\\ \langle Pref_2(x), s\rangle &\leftarrow HasKM(x,k), s = ls(13000, 17000)(k)\\ \langle Buy(x,p), s\rangle &\leftarrow \langle Pref_1(x,p), s_p\rangle, \langle Pref_2(x_k), s_k\rangle, s = 0.7 \cdot s_p + 0.3 \cdot s_k\end{aligned}$$

ID	MODEL	PRICE	KM
455	MAZDA 3	12500	10000
34	ALFA 156	12000	15000
1812	FORD FOCUS	11000	16000
⋮	⋮	⋮	⋮

- Problem: All tuples of the database have a score:
  - We cannot compute the score of all tuples, then rank them. Brute force approach not feasible for very large databases
- Top-k fuzzy LP problem: Determine efficiently just the top-k ranked tuples, without evaluating the score of all tuples. E.g. top-3 tuples

ID	PRICE	SCORE
1812	11000	0.6
455	12500	0.56
34	12000	0.50

Rule Languages and Semantic Web

- There are quite many LP/ASP systems (monotone/non-monotone)
  - each with its own feature set
  - some with SW interface
    - SWIProlog, DLV, ...
- More SW related: various frameworks exist ...
  - SWRL: rules with concept and role expressions as atoms
  - Datalog$^{\pm}$: Datalog with existential restriction on rule head
  - RuleML: quite large range of features
- The development of fuzzy LPs is essentially in parallel with that of classical LPs (since early '80s)
- A common problem with LP frameworks (incl. fuzzy)
  - Lack of standardised language and semantics
  - SWRL, RuleML are exceptions

Annotation domains & Datalog

- For Datalog, it it is like for RDFS
- The reasoning decision problems' complexity is inherited from their fuzzy variants. Decidable if lattice and truth space are finite, else undecidable in general

Conclusions

Conclusions & Future work

- We've overviewed basic concepts related to Fuzzyness in Semantic Web Languages, such as
  - RDFS, OWL 2, Datalog
- Semantic Web Applications:
  - Robotics, Ontology Mappings, Multimedia Object annotation, Matchmaking, (Multimedia/Distributed) Information Retrieval, Recommender Systems, User Profiling, . . .

Summary within Fuzzy Semantic Web Framework (IMHO)

Language	Mature Systems	Inference Algorithms	Query Answering
RDFS			
OWL 2			
OWL QL			
OWL EL			
OWL RL			
Rule Languages			

Emerging Field for SWLs: Enhanced Fuzzy Queries

- Fuzzy Quantified queries may provide many opportunities to improve CQ query features for any SWL: e.g.
  - *Visualise roads in which many of the recent car incidents involved severely injured people*
  - Fuzzy quantified query schema:

    $$Q \text{ of } B\,X \text{ are } A$$

  - Q is a fuzzy quantifier, e.g. *many*
  - $B\,X$ is a reference fuzzy set over which Q quantifies, e.g. *recent* (B) *car incidents* (X)
  - A is a fuzzy set imposing a condition to be satisfied, e.g. *severely injured people*
- Fuzzy Queries may be applied both to crisp ontologies as well as to fuzzy ontologies

That's it !