UNIVERSITY OF ABERDEEN

Charting the Potential of Description Logic for the Generation of Referring Expression

SELLC, Guangzhou, Dec. 2010

Yuan Ren, Kees van Deemter and Jeff Z. Pan

Department of Computing Science
University of Aberdeen, UK

Background

- Generation of Referring Expressions (GRE) algorithms *identify a target* referent:
- Express info known to be
  true of target
  false of all else

- But how does GRE model knowledge?

Knowledge Representation in classic GRE

- Sets of <Attribute,Value> pairs, e.g.
  - <Type, Poodle>
  - <Colour, Grey>
- These are *atomic* facts. Can't say
  - **All** poodles are grey
  - Kees is Dutch **or** Belgian
  - **If** x is part of y
and y is part of z
then x is part of z

Modern KR can do all this (and more)

- KL-One and semantic nets
- Modern descendants: Conceptual Graphs and Description Logic
  - Represent complex knowledge
  - Perform efficient automatic deduction

- Why not use modern KR for GRE?
- This talk: Description Logic (DL)
  - DL/OWL is now the language of the semantic web

Advantages

- A richer model of reference
  - New targets become identifiable
- Re-use
  - existing algorithms
  - existing ontologies
  - deduction for proving uniqueness
- Info represented *succinctly*, e.g.
  - *a* part of *b*, *b* part of *c*, *c* part of *d*, …
  - **For all** *x,y,z*: **if** *x* part of *y* **and** *y* part of *z* **then** *x* part of *z*

Remainder of this talk

- Introduction to DL
- Areces et al. on DL and GRE
- Extending the expressivity
- Extending the algorithm

Caveat: few empirical claims

This talk is about *what's possible* in GRE

Description Logics?

- A family of logic-based KR formalisms
- Describe domain in terms of concepts (classes), roles (relations) and individuals
- Smallest propositionally closed DL is **ALC** (**equivalent to $K_{(m)}$**)
  - Concepts constructed using ∪ , ∩, ∀, ∃

  E.g., domain elements that have a child who is a Doctor:

$$(\exists hasChild.Doctor)^{\mathcal{I}} = \{x | \exists y. \langle x, y \rangle \in hasChild^{\mathcal{I}} \text{ and } y \in Doctor^{\mathcal{I}}\}$$

DL Knowledge Base

- A TBox is a set of "schema" axioms (sentences), e.g.:

  { Dog ⊂ Animal, Dog ⊂ ¬ Woman
  hasFather ∘ hasBrother ⊂ hasUncle, feed ∘ love ⊂ feed }

  – i.e., a **background theory**

- An ABox is a set of "data" axioms (ground facts), e.g.:

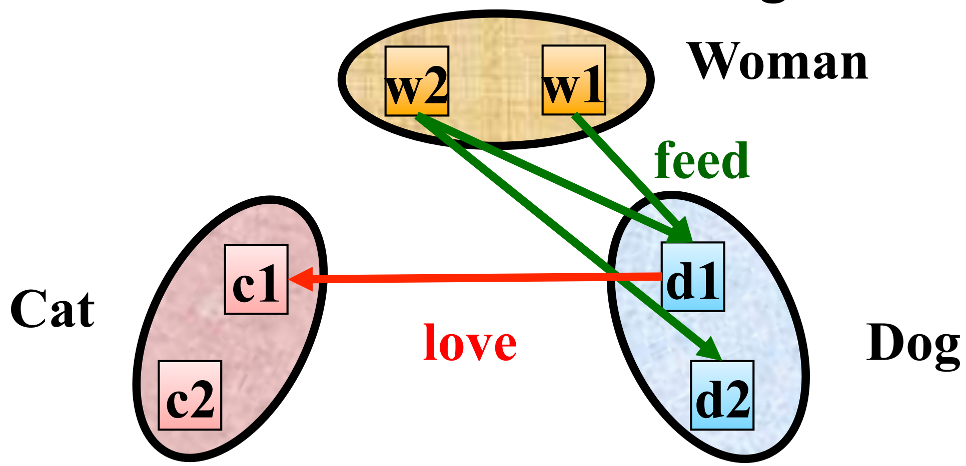
  {d1:Dog, w1:Woman, c1:Animal,
  (w1,d1):feed, (d1,c1):love
  }

- An NBox (Negation as Failure Box) is a set of "complete" concepts and properties e.g.:

  {Dog, Animal}

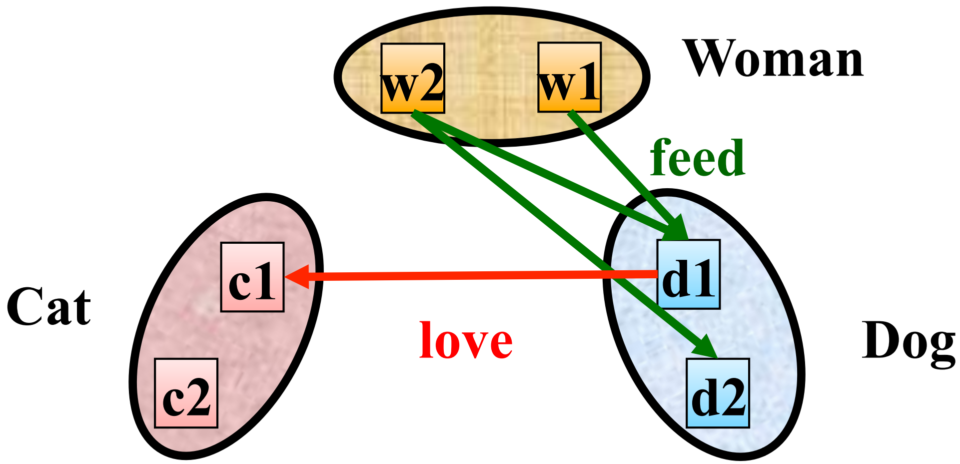
GRE example

- Uniquely identifying an object in context.
  - d1: the Dog that loves some Cat
  - w2: the Woman who feeds a Dog that doesn't love a Cat



From the DL point of view

- ABox assertion axioms
  - w1:Woman
    w2:Woman
  - d1:Dog, d2:Dog
  - c1:Cat, c2:Cat
  - (w1,d1):feed
    (w2,d2):feed
    (w2,d1):feed
  - (d1,c1):love



DL for GRE: the story so far

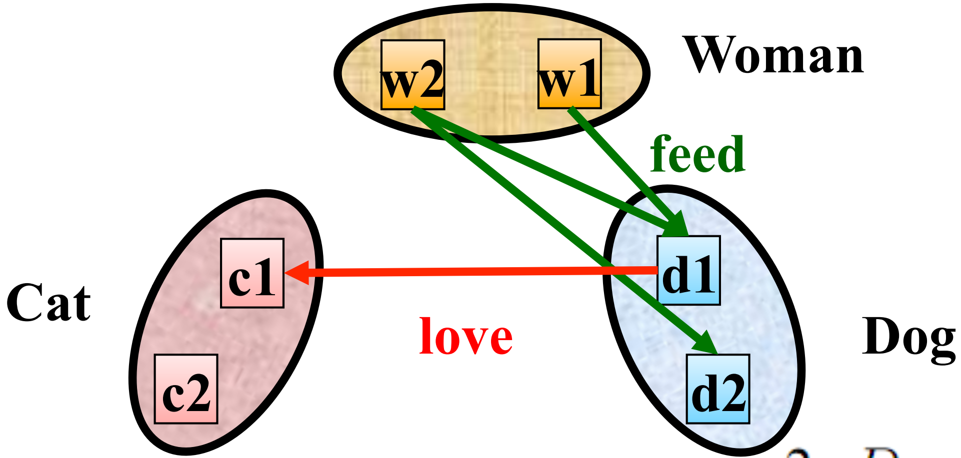
- Areces et al. (2008) re-interpret GRE as a problem of computing ALC formulas.

$\mathcal{L}$-GRE PROBLEM	
Input:	A model $\mathcal{M}$ and a target set $A \subseteq \Delta$.
Output:	A formula $\varphi \in \mathcal{L}$ such that $\|\varphi\| = A$ (if such a formula exists).

- An algorithm to compute distinguishing REs (if one exists) for all objects:
  - Generate & Test strategy
  - Start from atomic concept names, then extend with (negative) existential quantifier.

DL for GRE example

- Referring Expression as a DL formula



1. $Dog = \{d1, d2\}$,
   $\neg Dog \sqcap Woman = \{w1, w2\}$,
   $\neg Dog \sqcap \neg Woman = \{c1, c2\}$.

2. $Dog \sqcap \exists love.(\neg Dog \sqcap \neg Woman) = \{d1\}$,
   $Dog \sqcap \neg \exists love.(\neg Dog \sqcap \neg Woman) = \{d2\}$.

3. $(\neg Dog \sqcap Woman) \sqcap \exists feed.(Dog \sqcap \neg \exists love.(\neg Dog \sqcap \neg Woman)) = \{w2\}$,
   $(\neg Dog \sqcap Woman) \sqcap \neg \exists feed.(Dog \sqcap \neg \exists love.(\neg Dog \sqcap \neg Woman)) = \{w1\}$.

Logical foundations of GRE

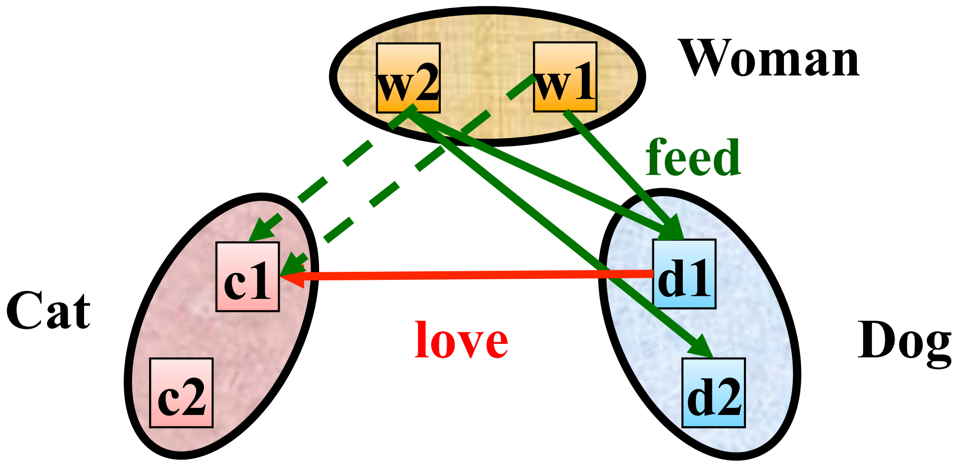
DL forces us to make these explicit, allowing subtle distinctions

- GRE relies on a Unique Name Assumption (UNA)
  - If Dog={d1,d2} but d1=d2 then Dog refers uniquely
  - In DL, we can use UNA, or write d1=/=d2, etc.

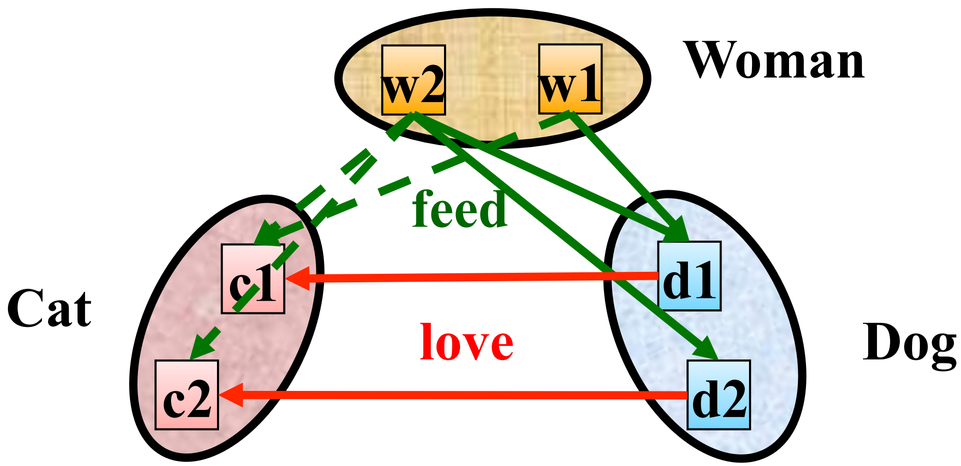
- GRE relies on a closed world assumption (CWA)
  - In the earlier example, there are no more than two dogs, no more than two women, etc.
  - Without this, we would never be sure of uniqueness
  - DL: we can localise CWA, using an NBox

ABox Not Enough

- Tbox:
  - Woman ⊂ ¬Dog
  - Dog ⊂ ¬ Cat
  - Cat ⊂ ¬ Woman
  - T ⊂ Dog ∪ Woman ∪ Cat
  - feed ∘ love ⊂ feed
- Nbox:
  - {Dog,Woman,Cat,love,feed}



Additional Quantifiers



- Now Areces et al. cannot identify any of the 6 objects
- Additional quantifiers make them referable
  - c1: The cat which is fed by (at least) 2 women
  - w1: The woman feeding only those fed by at least 2 women
  - w2: The woman who feeds all dogs

Representing Quantifiers in Ontologies

We can use Generalised Quantifiers, e.g.,

"only", "five" *(5)*, "at least two" *(≥2)*

Example:

- English: *"The woman who loves at least two dogs"*
- Set theory + GQ: *{y ∈ Woman: **≥2** (Dog,{z: Love(y,z)})}*

But which **quantifiers** exactly?

Proposal: use the numerical quantifiers

Table 1: Expressing GQ in DL

	QAB	DL
1	$\geq nN2\{z : Ryz\}$	$y :\geq nR.N2$
2	$\geq nN2\neg\{z : Ryz\}$	$y :\geq n\neg R.N2$
3	$\geq n\neg N2\{z : Ryz\}$	$y :\geq nR.\neg N2$
4	$\geq n\neg N2\neg\{z : Ryz\}$	$y :\geq n\neg R.\neg N2$
5	$\leq nN2\{z : Ryz\}$	$y :\leq nR.N2$
6	$\leq nN2\neg\{z : Ryz\}$	$y :\leq n\neg R.N2$
7	$\leq n\neg N2\{z : Ryz\}$	$y :\leq nR.\neg N2$
8	$\leq n\neg N2\neg\{z : Ryz\}$	$y :\leq n\neg R.\neg N2$

Example:

N2 -- Dog
R -- Love

- Quantifiers of Type 1,
  e.g., n = 1: the *existential* quantifier
- Quantifiers of Type 7,
  e.g., n = 0: the *only* quantifier
- Quantifiers of Type 6,
  e.g., n = 0: the *all* quantifier

Generating SROIQ-enabled REs

- GROWL: a GRE algorithm using OWL-2
  - Generate-and-test strategy
  - Using DL reasoning
  - Generating increasingly complex descriptions
    - Complexity measured by the structure of the expressions
      - 1 Dog
        complexity of (negated) atomic concept is 1
      - 2 $\neg Dog \sqcap Woman$
        complexity of conjunction (disjunction) is the maximal complexity of conjuncts (disjuncts) +1
      - 4 $Dog \sqcap \neg \exists love.(\neg Dog \sqcap \neg Woman)$
        complexity of existential (universal) restriction is the complexity of filler + 1

Generating SROIQ-enabled REs

- GROWL: a GRE algorithm using OWL-2
  - Starting from the basic terms, such as
    - Names (e.g. Dog, feed)
    - Inverse of named roles (such as $feed^-$)
    - Negation of another basic term
      (e.g., ¬Dog, ¬feed, ¬$feed^-$)
  - For each existing term / description, create new description by
    - Extending with conjunction/disjunction
    - Extending with existential/universal restriction
    - Extending with cardinality restriction
  - Until no new non-empty description can be created

Generating SROIQ-enabled REs

- Deciding whether a new description should be accepted
  - Accept if no existing description has same extension
  - But still non-empty
- Using standard reasoning services
  - Concept subsumption checking
  - Concept satisfiability checking

Algorithm A-2: $Add(D, e)$

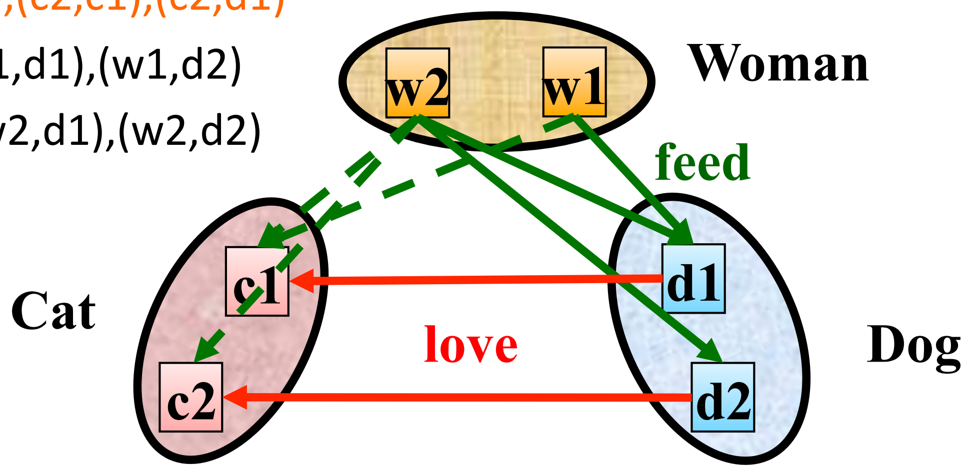
INPUT: D, e

OUTPUT: (Extended)Description Queue D

```
1: for d ∈ D do
2:     if Ins(d) = Ins(e) then
3:         return D
4: if Ins(e) ≠ ∅ then
5:     D := D ∪ {e}
6: return D
```

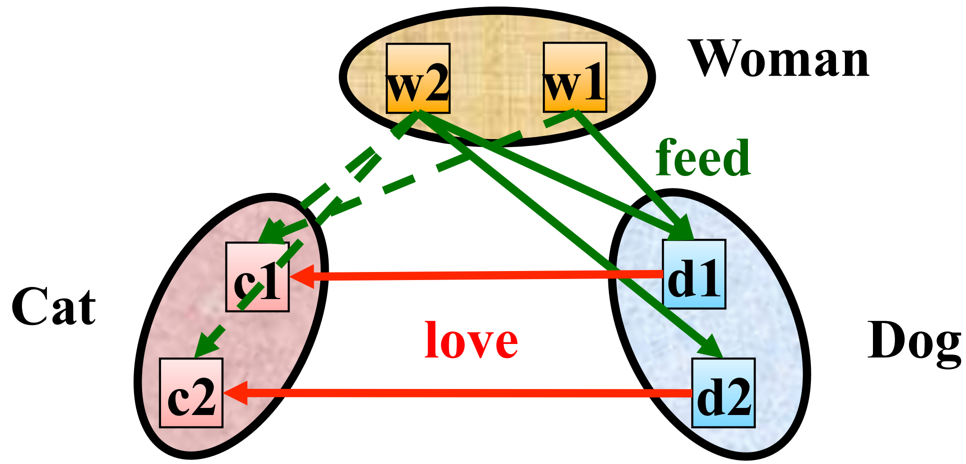
Example Revisited

- Starting from basic terms
  - D={Woman, Dog, Cat, ¬ Woman, ¬ Dog, ¬ Cat, feed, feed$^-$, ¬feed$^-$ etc.}
  - Ins(Woman) = {w1,w2}
  - Ins(Dog) = {d1,d2}
  - Ins(Cat) = {c1,c2}
  - Ins(¬Woman) = {c1,c2,d1,d2}
  - …
  - Ins(feed) = {(w1,d1),(w1,c1), (w2,d1), (w2,c1),(w2,d2),(w2,c2)}
  - Ins(feed$^-$)= {(d1,w1),(c1,w1), (d1,w2), (c1,w2),(d2,w2),(c2,w2)}
  - Ins(¬feed$^-$) = {(d1,d2),(d1,c1),(d1,c2),(d1,d1),(d2,d1),(d2,d2),(d2,c1),(d2,c2),(d2,w1) (c1,c1),(c1,c2),(c1,d1),(c1,d2),(c2,c2),(c2,c1),(c2,d1) (c2,d2),(c2,w1),(w1,w1),(w1,w2),(w1,d1),(w1,d2) (w1,c1),(w1,c2),(w2,w1),(w2,w2),(w2,d1),(w2,d2) (w2,c1),(w2,c2),}



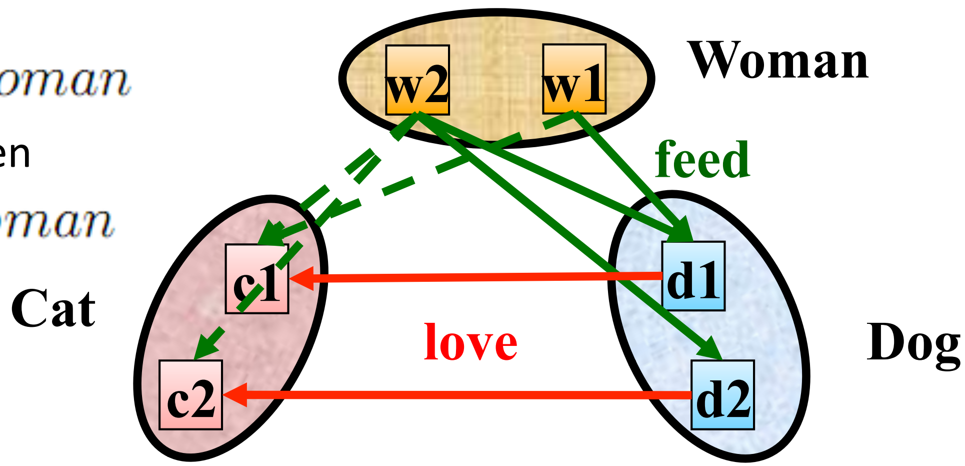
How to Refer to c1?

- Extending Woman with maximal Cardinality restriction
  <=0(¬feed$^-$).Woman (things that are fed by all women)
- Ins(<=0(¬feed$^-$).Woman) = {**c1,d1**}. This is non-empty and not identical to any existing description
- Therefore, the new concept is added to D:
- D = {Woman, Dog, Cat, ¬Woman, ¬Dog, ¬Cat, **<=0(¬feed$^-$). Woman**, etc.}
- Conjoining this with Cat:
  - Ins(Cat **&** <=0(¬feed$^-$).Woman) = {**c1**}, c1 identified!



All objects become referable

- w1: the woman that does not feed all cats;
  $\{w1\} = Woman \sqcap \exists \neg feed.Cat$
- w2: the woman that feeds all cats;
  $\{w2\} = Woman \sqcap \leq 0 \neg feed.Cat$
- d1: the dog that is fed by all women;
  $\{d1\} = Dog \sqcap \leq 0 \neg feed^{-}.Woman$
- d2: the dog that is not fed by all women;
  $\{d2\} = Dog \sqcap \exists \neg feed^{-}.Woman$
- **c1: the cat that is fed by all women;**
  $\{c1\} = Cat \sqcap \leq 0 \neg feed^{-}.Woman$
- c2: the cat that is not fed by all women
  $\{c2\} = Cat \sqcap \exists \neg feed^{-}.Woman$



Conclusion (1)

We advocate using modern KR

- background knowledge should be considered
- expressing complex as well as atomic info
- reusing reasoning algorithms

Conclusion (2)

- Specific contributions:
  - Using DL reasoning to infer implicit knowledge (i.e., computing non-asserted information)
  - Generating REs taking into account incomplete knowledge
  - Extending the expressivity of GRE to OWL-2 (i.e., using new quantifiers, such as ≥2feed.Dog)
  - Introducing Nbox for a partially closed world
- Open issues:
  - How useful are the newly generated REs?
  - How to choose the "best" RE?

Conclusion (2)

- Choosing the numerical quantifiers, have we used the "best" class?
- Note: GROWL generates descriptions of sets as well as individuals
- Description of sets will require new quantifiers
  - The dogs fed by between 3 and 7 women
- Some of these may even be non-first order, e.g.,
  - The dogs fed by most women
  - The dogs fed by an even number of women
- In finite domains, these were already expressible (though maybe less succinctly)