Translating Rules in Natural Language to RuleML

Adam Wyner
Department of Computer Science

August 27, 2012
RuleML 2012
Montpellier, France
Overview

• Problem statement and SOA.
  – From natural language expressions to formal, processable language.

• Tools:
  – Attempto Controlled English and C&C/Boxer.
  – Translation samples.

• Relation to:
  – SymposiumPlanner
  – Semantics of Business Vocabulary and Rules

• Problems and progress.
The Known Problem

• People use natural languages for communication, knowledge representation, and reasoning.
• Machines use formal languages, knowledge representation, and reasoning. RuleML is a particular example.
• How to bridge between natural and formal languages - the knowledge acquisition bottleneck?
Suggested Solutions

• Don't bother with natural language because:
  – context dependent, ambiguous, subjective, artifact, variable constructions and meanings, arbitrary meaning, complex, irony, sarcasm, unrestricted, when I was at school, I know language....

• Apply mathematical tools to analyse language:
  – Categorial grammar (Bar Hillel 1950s)
  – Phrase structure grammars (Chomsky 1950s)
  – Formal semantics (Montague 1960s)
  – Discourse analysis (Kamp/Heim 1980s)

Developments in the formal maths approach:
- van Bentham and ter Meulen (1997) *Handbook of Logic and Language*.

Developments in rule-based (or combo) computational approaches, where we parse an input sentence, output a semantic representation, draw inferences.
- Blackburn and Bos (2005) *Representation and Inference for Natural Language*.
State-of-Affairs - Industrial

• Industrial scale applications with *Oracle Policy Automation*: Scope legislation and regulation, express policies and knowledge base in a "just enough" controlled natural language, served as web-based business/policy system.
  – Creating Rules in Oracle Policy Automation
  – UK DirectGov Benefits Calculator

• This is one of the industrial competitors with respect to rules representation and reasoning using web-based natural language interfaces.

• Why no similar open-source, academic tool?
State-of-Affairs – Open NL Tools

- Attempt to Controlled English.
- C&C/Boxer.

- Discuss each, give examples of parses and semantic interpretations, and show translations to RuleML.

- Discuss issues.
The Attempto Project
   – http://attempto.ifi.uzh.ch/site/
Thanks to Tobias Kuhn, Kaarel, and Norbert Fuchs.
A 'looks like English' formal language.
Fixed (but large and extensible) lexicon and constrained grammar.
Written as Prolog Definite Clause Grammar.
Well-formed expressions are parsed and translated to Discourse Representation Structures (FOL).
Deterministic (no ambiguity), assisted by construal rules.
Translation to RuleML.
Issues of deep semantic interpretation.
Useful for building consistent knowledge bases; can be used in conjunction with an inference engine.
ACE Tools

• APE, the ACE parser. Input a sentence that is well formed in ACE, the output is a parse and semantic interpretation. Fails if it cannot parse.

• ACE Editor, which has a look ahead editor that guides the user to input grammatically well-formed sentences (allows semantically ill-formed expressions).

• ACE View, a Protege plugin NL interface to create, view, edit, and query OWL ontologies and SWRL rulesets.

• RACE, an ACE reasoner to check consistency of statements, prove theorems, and answer queries.
ACE Constructions

• negation on nouns or verbs, conjunction, disjunction, conditionals, quantifiers, adjectives, relative clauses, discourse anaphora, modals ("necessity", "permission"), possessives, prepositional phrases, verbs with three arguments, verbs with subordinate clauses ("know"), negation-as-failure ("It is not provable...") and so on.

• DRS for interclausal and intersentential constructions to bind variables:
  – If a customer is rich, then the customer gets a discount.
  – If a customer is rich, then she gets a discount.
  – A customer is rich. She gets a discount.

• DRS for rhetorical relations: presupposition, explanation, justification, ....
ACE Editor Example

ACE Text Editor

...
ACE Editor Example

ACE Text Editor

every ...

text

function word + noun + adjective

more
most

new...
account
address
age
aircraft
airline
ancestor
animal
ape
apple
approach
article
asset
ball
bank
bed

active
angrier
angriest
angry
authenticated
automatic
average
bad
best
better
big
bigger
biggest
blue
bluer

< Delete
OK Cancel
ACE Editor Example

every big bank ...

function word
- can
- cannot
- does not
- has to
- is
- is not
- must
- of
- should
- that
- which

intransitive verb
- new...
- ages
- appears
- arrives
- barks
- blinks
- boils
- collapses
- comes
- counts
- dances
- drinks
- drives
- eats
- exists
- expires

transitive verb
- new...
- accepts
- accesses
- assigns
- awaits
- beats
- believes

new variable
- X
- X1
- X2
- Y
- Y1
- Y2
- Z

OK
Cancel
ACE Editor Example

ACE Text Editor

every big bank cannot collapse before ...

text

function word
a
an
at least
at most
every
everybody
everything
exactly
less than
more than
no
nobody
nothing
somebody
something
what

proper name
new...
Austria
Berlin
Bill
Canada
France
Germany
Italy
Jane
John
Lara
Mary
Mike
Paris
Rome
Russia

new variable
X
X1
X2
Y
Y1
Y2
Z
Z1
Z2

reference
the bank
itself
ACE Editor Example

every big bank cannot collapse before every institution ...

text

function word + preposition + new variable

and
more
most
of
or
that

new...
aboard
about
above
across
after
against
along
alongside
amid
among
amongst
around
as
at
before

< Delete

OK Cancel

27/08/12

Wyner, RuleML, (cc) by-nc-sa license
ACE Syntax Tree

every big bank can not collapse before every institution.
ACE DRS

Pretty-Printed DRS

[]

[A]
object(A,bank,countable,na,eq,1)-1/3
property(A,big,pos)-1/2
=>
[]

NOT
[]

CAN
[]

[B]
object(B,institution,countable,na,eq,1)-1/9
=>

[C]
predicate(C,collapse,A)-1/6
modifier_pp(C,before,B)-1/7
ACE DRS XML

```xml
<DRS domain="">
  <Implication>
    <DRS domain="A">
      <Object
        ref="A"
        noun="bank"
        struct="countable"
        unit="na"
        numrel="eq"
        numa="1"
        sentid="1"
        tokid="3"/>
    </DRS>
  </Implication>
  <property ref="A" adj="big" degree="pos" sentid="1" tokid="2"/>
</DRS>

<DRS domain="">
  <Negation>
    <DRS domain="">
      <Possibility>
        <DRS domain="">
          <Implication>
            <DRS domain="B">
              <Object
                ref="B"
                noun="institution"
                struct="countable"
                unit="na"
                numrel="eq"
                numa="1"
                sentid="1"
                tokid="9"/>
          </DRS>
        </Implication>
        <Predicate ref="C" verb="collapse" subj="A" sentid="1" tokid="6"/>
        <Modifier_pp ref="C" prop="before" obj="B" sentid="1" tokid="7"/>
      </Possibility>
    </DRS>
  </Negation>
</DRS>
</Implication>
</DRS>
```
forall (A,=> (& (object(B,A,bank,countable,na,eq,1)−1/3, property(B,A,big,pos)−1/2), 
−(exists (C,&(accessibility_relation(B,C)−accessibility_relation/0, 
forall(D,=> (object (C,D,institution,countable,na,eq,1)−1/9, 
exists (E, & (predicate(C,E,collapse,A)−1/6,modifier_pp(C,E,before,D)−1/7))))))))
ACE RuleML

```xml
<RuleML>
  <Assert>
    <Forall>
      <Var>_G431</Var>
      <Implies>
        <Atom>
          <Rel>object</Rel>
          <Var>_G431</Var>
          <Ind>customer</Ind>
          <Ind>countable</Ind>
          <Ind>na</Ind>
          <Ind>eq</Ind>
          <Data>1</Data>
        </Atom>
      </Implies>
    </Forall>
    <Exists>
      <Var>_G1557</Var>
      <Var>_G1280</Var>
      <And>
        <Atom>
          <Rel>property</Rel>
          <Var>_G1557</Var>
          <Ind>important</Ind>
          <Ind>pos</Ind>
        </Atom>
        <Atom>
          <Rel>predicate</Rel>
          <Var>_G1280</Var>
          <Ind>be</Ind>
          <Var>_G431</Var>
          <Var>_G1557</Var>
        </Atom>
        <And>
        </And>
      </Exists>
    </Implies>
  </Forall>
</Assert>
</RuleML>
```
ACE Query

**Axioms:** Every man is a human. Every woman is a human. Mary is a woman. John is a man. Jill is a woman.

**Query:** Who is a woman and is a human?

**Parameters:**
• Subset 3
  - 2: Every woman is a human.
  - 3: Mary is a woman.
  - Substitution: who = Mary

• Subset 4
  - 2: Every woman is a human.
  - 5: Jill is a woman.
  - Substitution: who = (at least 1) human

• Subset 5
  - 2: Every woman is a human.
  - 5: Jill is a woman.
  - Substitution: who = (at least 1) woman

• Subset 6
  - 2: Every woman is a human.
  - 5: Jill is a woman.
  - Substitution: who = Jill
Every woman is a human. Mary is a woman. Jill is a woman.

Mary is a woman.
Jill is a woman.
Mary is a human.
Jill is a human.
Applications

- SymposiumPlanner Q&A.
  - http://ruleml.org/SymposiumPlanner/
- Building OWL ontologies and SWRL rules in Protege.
  - http://protege.stanford.edu/
SymposiumPlanner

The Query in ACE  ➔ Which papers are full and accepted?

APE +
ACE2RML
Translator

The RRuleML Message  ➔

XSLT Stylesheet

The Message in Prova  ➔

<Message directive="query">
  <oid><Ind>RuleML-2011-IJCAI</Ind></oid>
  <protocol><Ind>esb</Ind></protocol>
  <sender><Ind>User</Ind></sender>
  <receiver><Ind>RuleML-2011-IJCAI</Ind></receiver>
  <content>
    <Atom>
      <Rel>getPapers</Rel>
      <Ind>full</Ind>
      <Ind>accepted</Ind>
      <Var>B</Var>
    </Atom>
  </content>
</Message>

[httpEndpoint:3,esb,httpEndpoint,query, [getPapers,full,accepted,<2901>]].
Question and DB Input

• Which papers are full and accepted?

DB that can be queried:

• "RuleML for Farming" is a paper.
• "RuleML for Farming" is full.
• "RuleML for Farming" is accepted.
• "RuleML for Beer Brewing" is a paper.
• "RuleML for Beer Brewing" is full.
• "RuleML for Beer Brewing" is accepted.
• "RuleML for Biomolecular Processes" is a paper.
• "RuleML for Biomolecular Processes" is accepted.
• "RuleML for Chemical Manufacture" is a paper.
ACE Problems

• Input well-formed sentence that is meaningful to the user, but the semantic interpretation is not quite what one wants.
• Wyner, van Engers, and Bahreini (2010) From Policy-making Statements to First-order Logic
• Input well-formed 'junk', since there is not enough semantic control between words – entered in ACE Editor.
  – A more bad and most authenticated address cannot boil extremely alongside no declaration that ages amongst Austria.
• Declarative or interrogative sentences are third person and simple present tense.
• No thematic roles, no tenses, no aspects,....
• Moral is that if you are well-behaved and pay close attention, it will help you.
Other Examples

• Extracted sentences from Hirtle's thesis, RuleML Primers, RuleML slides.

• Not everything is well-formed and/or unproblematic in ACE.
  – Peter Miller's spending has been min 5000 euro in the previous year.
  – Peter Miller spent at least 5000 euros in the previous year.

Gerund, where a verb appears in a nominal form.
  – The discount for a customer buying a product is 5 percent if the customer is premium and the product is regular.
  – If a customer is premium and a product is regular, then the discount for the customer buying the product is 5 percent.

Not parseable for lexical issues and punctuation. Clause order issues.

• And other issues....
Other Examples

• Sentences tested using a local installation of ACE.
  – If a customer is a student then she receives a discount of five percent or the customer receives 200 dollars.
  – It is necessary that a customer is satisfied.
  – Mary knows that a customer should be satisfied.
  – Mary sends the check to Bill.
  – Mary sends the check to Bill on Tuesday.
C&C/Boxer

- C&C/Boxer
- GMB Webdemo
  - https://urd.let.rug.nl/basile/gsb/webdemo/demo.php
- Groningen Meaning Bank
  - http://gmb.let.rug.nl/
- Thanks to Valerio Basile and Johan Bos.
C&C/Boxer

- Lexically given grammar (how words combine).
- Uses a statistical model to output the most likely parse.
- Successful wide-coverage (of the Penn Treebank corpus). Parses and semantically represents more than ACE.
- Translates to DRS (FOL).
- Nutcracker inference tool (textual entailment th).
- Translation to RuleML.
- Issues of interpretation increased.
C&C/Boxer Example

• David Cameron is a British citizen.
• The woman gave a check to a man

• https://urd.let.rug.nl/basile/gsb/webdemo/demo.php

• Examples with or without thematic roles or tense. No thematic roles in RuleML – 'flat' predicate argument structure.
• The woman should have given a check to the man

\[ \exists A : \exists B : \exists C : \exists D : ( n1\text{woman}(A) \land ( n1\text{check}(B) \land ( v1\text{give}(C) \land ( r1\text{agent}(C,A) \land ( r1\text{patient}(C,B) \land ( n1\text{man}(D) \land r1\text{to}(C,D) ) ) ) ) ) ) \]

RuleML version removes temporal and event arguments as well as temporal relations. Event argument remains as bound variable. Leaves thematic roles.
Other Examples Tested

• Reaction RuleML
  – A customer gets a discount of 7.5 percent to be deducted from each purchase of a product, starting on July 1st, 2012 and ending on December 31st, 2012, if the customer is premium and the product is luxury.

• RuleML Inc Discount
  – RuleML Inc discount contract: According to this contract, which is applicable in all member states of the European Union, a RuleML Inc customer gets a discount of 7.5 percent to be deducted from each purchase of a RuleML Inc product starting on July 1st, 2012 and ending on December 31st, 2012 if the customer is premium and the product is luxury.

• Citizen
  – A British overseas territories citizen who falls to be treated as a national of the United Kingdom for the purposes of the Community Treaties shall be entitled to be registered as a British citizen if an application is made for his registration as such a citizen.

• Vehicle
  – Vehicles are not permitted in this park.
Modals

• Alethic (necessity, ....), Epistemic (know, ....), Temporal, Deontic (must, may, ....).

• Subclasses of modal operators and propositional attitudes with different inference properties. Why are these not relations rather than sentential operators?
  – Factive – know, saw, heard,...
  – Non-factive – believe, remember (?),...
  – Contrafactive – wish, imagine, ...
  – Epistemic-root interpretation – may, must, can,....
  – Scope issues with de re/de dicto.
Issue

- Representational correspondence and/or redundancy.
- Example, NL semantics uses event arguments to tie together the representation of terms with respect to a predicate, not flat, labelled arguments (compare to OrdLab graphs).
- Different typing of expressions (data, individual, relation).
- How to handle Production, Trigger, and ECA rules? Some lexical, some structural?
Using SBVR in C&C/Boxer

• Semantics of Business Vocabulary and Rules.
  – http://www.omg.org/spec/SBVR/1.0/
• Issues about statutory and judicial interpretation.
• Interpretive guidelines to resolve vagueness, ambiguity, contexts of use.
• There is literature and practice guidelines (e.g. UK and USA) about how legislation ought to be written to avoid problematic interpretation:
  – Solan (2010) *The Language of Statues: Laws and Their Interpretation*
Comments on Interpretation

- ACE and C&C/Boxer parse text (deterministically or statistically).
- Given a parse, we get a translation to a semantic representation.
- The tools are neutral about vagueness and interpretation. Some may make lexical semantic interpretations.
- They do contribute to issues bearing on a formal semantic representation, e.g. quantifier or negation scope, binding, subordination. There are some instances where someone is 'hung upon a comma', but Solan argues this is relatively rare.
- Interpretation as refinement or embedding.
- Perhaps have multi-parsing tools to maintain ambiguity.
  - Quelo: a NL-based Intelligent Query Interface
Future work

- Close evaluation of parses, semantic translations, and conversion to RuleML.
- Compare and contrast outputs.
- Extension and application of ACE and GMB. For example, extract the vocabulary and structures from SBVR or see what is/is not available in ACE or GMB.
- Corpus development.
- Gold standards for ACE or GMB.
- Applications
  - Ontologies
  - Knowledge bases
  - Business and policy rule systems
Acknowledgements

- Thanks to RuleML organisers for the invitation.
- Thanks for support to FP7-ICT-2009-4 Programme, IMPACT Project, Grant Agreement Number 247228.
- Thanks to colleagues who developed ACE and C&C/Boxer.
Thanks for your attention!

• Questions?
• Contacts: Adam Wyner adam@wyner.info
• Join Adrian Paschke and I for further discussion of Natural Language Interfaces to RuleML over dinner in a downtown Montpellier restaurant at 20:30. Location TBA.