At Last! A Reason to Generate Language from Logic

Robert Dale
rdale@ics.mq.edu.au

Collaborative work with Dave Barker-Plummer, Stanford; Richard Cox, U Sussex; Mark Dras and Rolf Schwitter, Macquarie U
The Aims of This Talk

- To introduce a new problem in Natural Language Generation
- To sketch the approach we intend to take
- To provide some initial data analysis
Agenda

• Approaches to Generation, Past and Present
• The OpenProof Project
• Paraphrase Selection
• A Look at Some Real Data
• Next Steps
How Natural Language Generation Used To Be Done

The predominant approach until this decade:

• Requires a rich input knowledge representation
• Discourse generation starts with a communicative goal
• Makes subtle linguistic decisions about what to say and how to say it using a domain model, a discourse model and a user model
A Traditional NLG Architecture

Document Planning

Micro Planning

Surface Realization

-content determination
-lexicalisation
-aggregation
-referring expression generation
-syntax, morphology, orthography and prosody

text structuring
One Example:
An SPL input to KPML

```
(l / greater-than-comparison
  :tense past
  :exceed-q (l a) exceed
  :command-offer-q not commandoffer
  :proposal-q not proposal
  :domain (m / one-or-two-d-time :lex month :determiner the)
  :standard (a / quality :lex average :determiner zero)
  :range (c / sense-and-measure-quality :lex cool)
  :inclusive (r / one-or-two-d-time
    :lex day
    :number plural
    :property-ascription (r / quality :lex rain)
    :size-property-ascription
      (av / scalable-quality :lex the-av-no-of)))
```

The month was cooler than average with the average number of rain days.
Decision Making in a Systemic Network
Realisation Statements

- **Passive**
  - Insert Passive
  - Classify Passive BeAux
  - Insert PassParticiple
  - Classify PassParticiple EnParticiple

- **Active**

- **Agentive**
  - Insert Agent
  - Insert Actor
  - Preselect Actor Nominal Group
  - Conflate Actor Agent
  - Insert AgentMarker
  - Lexify AgentMarker by
  - Order AgentMarker Agent

- **Agentless**
How Natural Language Generation Gets Done Today

- Input is either:
  - an underspecified knowledge representation
  - other texts
- Language models are used to choose most likely realisation
Problems

- For the earlier approaches:
  - The rich underlying representations just don't exist
- For the later approaches:
  - No insights into the really interesting questions about language use
Agenda

• Approaches to Generation, Past and Present
• The OpenProof Project
• Paraphrase Selection
• A Look at Some Real Data
• Next Steps
Language, Proof and Logic
A Translation Exercise

7.12 (Translation) Translate the following English sentences into FOL. Your translations will use all of the propositional connectives.

1. If $a$ is a tetrahedron then it is in front of $d$.
2. $a$ is to the left of or right of $d$ only if it’s a cube.
3. $c$ is between either $a$ and $e$ or $a$ and $d$.
4. $c$ is to the right of $a$, provided it (i.e., $c$) is small.
A Grade Grinder Report

EXERCISE-7.12.Sentences-7.12.error.1=*** Your first sentence, "FrontOf(a,d) → Tet(a)", is not equivalent to any of the expected translations.
The Grade Grinder Dataset

The Grade Grinder

- can process solutions to 489 of the 748 exercises in the LPL book
- has been used by more than 38000 individual students over the last eight years, from around 100 institutions in around a dozen countries
- has assessed approximately 1.8 million individual submissions (each of which can contain zero or more exercises)
Hypothesis

• Perhaps we can provide better feedback by translating the student's errored solution back into natural language, so they can see their error
An Example

• English sentence:
  – John is either at the library or at home.

• Incorrect student translation (too weak):
  – Lib(j) ∨ Home(j)

• Correct translation:
  – Lib(j) ∨ Home(j) ∧ ¬(Lib(j) ∧ Home(j))

• A possible back-translation of the student's answer:
  – John is either at home or at the library or both.
### What This Might Look Like

<table>
<thead>
<tr>
<th>You were asked to translate:</th>
<th>John is either at the library or at home.</th>
</tr>
</thead>
<tbody>
<tr>
<td>You translated this as:</td>
<td>Lib(j) \lor\ Home(j)</td>
</tr>
<tr>
<td>But what you said really means:</td>
<td>John is either at home or at the library or both.</td>
</tr>
</tbody>
</table>
Agenda

• Approaches to Generation, Past and Present
• The OpenProof Project
• Paraphrase Selection
• A Look at Some Real Data
• Next Steps
Generating Paraphrases

The Basic Idea:
• The same logical form can be rendered in many different ways in NL
• Some renderings may be easier for a student to understand
• Some renderings may make it easier for a student to see where they have gone wrong

The Aim:
• to develop automatic natural language paraphrase capabilities that, given a student’s incorrect answer, are able to select and formulate an appropriate natural language expression that makes clear the difference between this and the correct answer
Paraphrase 'Distance From Source'

\[ \text{Home(\text{john})} \lor \text{Home(\text{mary})} \land \neg[\text{Home(\text{john})} \land \text{Home(\text{mary})}] \]

- Either John is home or Mary is home and it’s not the case that John is home and Mary is home
- Either John or Mary is home and it’s not the case that John and Mary are both home
- Either John or Mary is home but it’s not the case that John and Mary are both home
- Either John or Mary is home but it’s not the case that both of them are home
- Either John or Mary is home but not both
A Paraphrase Graph

Subject reduction by predicate conjunction

Explicit contrast

Both introduction

Pronoun introduction

Ellipsis
Basic Ideas

• Paraphrase n is rewritten as Paraphrase m by a tree rewrite rule
• Rewrite rules have a cost, or cause a certain amount of damage (including information loss)
• Paraphrases have properties or effects: they emphasise certain things
• The further a paraphrase is from the literal rendering the harder it may be to see the relationship between logic and NL …
• … but literal renderings can be significantly more complex than the simplest NL rendering
Paraphrases #2

• $\forall x \forall y \forall z ((\text{FatherOf}(x,y) \land \text{FatherOf}(y,z)) \rightarrow \text{Nicer}(x,y)$
• For all $x$, $y$ and $z$, if $x$ is the father of $y$ and $y$ is the father of $z$ then $x$ is nicer than $y$
• For all $x$, $y$ and $z$, if $x$ is z’s paternal grandfather and $y$ is z’s father, then $x$ is nicer than $y$
• For all $z$, z’s paternal grandfather is nicer than z’s father
• It’s the case for everyone that their paternal grandfather is nicer than their father
Paraphrases #3: De Morgan’s Laws

• \( \neg (P \land Q) \iff \neg P \lor \neg Q \)
  – It’s not the case that both P and Q \iff Either not P or not Q
  – It’s not the case that both John and Simon are telling the truth
  – Either John isn’t telling the truth or Simon isn’t telling the truth

• Add ‘synonymy by negation’:
  – Either John is lying or Simon is
Contextual Constraints on Paraphrase Choice

What we know or might be able to infer:

• The specific mistake that has been made
• The extent to which the student is comfortable with other parts of the translation
• What concepts they are already comfortable with
• What mistakes they have made before

So:

• Learn the mapping from user model and task model to preferred paraphrase
Agenda

• Approaches to Generation Past and Present
• The OpenProof Project
• An Approach to Paraphrase Selection
  • Some Data Analysis
• Next Steps
Data Selection for Initial Exploration

- We computed the number of GG submissions per LPL exercise and rank ordered them; Exercise 7.12 from Chapter 7 (which introduces conditionals) was selected.
- 74,000 submitted solutions, of which 42,416 were erroneous (57%), containing 148,681 incorrect translation solutions.
- The solutions were submitted by 11,925 students representing an average of 12.47 erroneous sentences per student.
Exercise 7.12: Sentences 1-10

1. If a is a tetrahedron then it is in front of d.
2. a is to the left of or right of d only if it's a cube.
3. c is between either a and e or a and d.
4. c is to the right of a, provided it (i.e., c) is small.
5. c is to the right of d only if b is to the right of c and left of e.
6. if e is a tetrahedron, then it's to the right of b if and only if it is also in front of b.
7. If b is a dodecahedron, then if it isn't in front of d then it isn't in back of d either.
8. c is in back of a but in front of e.
9. e is in front of d unless it (i.e., e) is a large tetrahedron.
10. At least one of a, c, and e is a cube.
Exercise 7.12: Sentences 11-20

11. a is a tetrahedron only if it is in front of b.
12. b is larger than both a and e.
13. a and e are both larger than c, but neither is large.
14. d is the same shape as b only if they are the same size.
15. a is large if and only if it's a cube.
16. b is a cube unless c is a tetrahedron.
17. If e isn't a cube, either b or d is large.
18. b or d is a cube if either a or c is a tetrahedron.
19. a is large just in case d is small.
20. a is large just in case e is.
An Error Taxonomy

45 distinct error types organised under the following categories:

- Structural Errors
- Connective Errors
- Atomic Errors
  - Predicate Errors
  - Argument Errors
# Examples of Errors

<table>
<thead>
<tr>
<th>#</th>
<th>Reference solution</th>
<th>Errored solution</th>
<th>Type</th>
<th>Subtype</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( \text{Tet}(a) \rightarrow \text{FrontOf}(a,d) )</td>
<td>( \text{FrontOf}(a,d) \rightarrow \text{Tet}(a) )</td>
<td>1</td>
<td>Antecedent–Consequent Reversal</td>
</tr>
<tr>
<td>2</td>
<td>( \text{Tet}(a) \rightarrow \text{FrontOf}(a,d) )</td>
<td>( \text{FrontOf}(a,b) \rightarrow \text{Tet}(a) )</td>
<td>1</td>
<td>Antecedent–Consequent Reversal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3i</td>
<td>Incorrect Constant</td>
</tr>
<tr>
<td>3</td>
<td>( \text{Tet}(a) \rightarrow \text{FrontOf}(a,d) )</td>
<td>( \text{Tet}(a) \lor \text{FrontOf}(a,d) )</td>
<td>2</td>
<td>Disjunction for Conditional</td>
</tr>
<tr>
<td>4</td>
<td>( \lnot \text{Cube}(e) \rightarrow (\text{Large}(b) \lor \text{Large}(d)) )</td>
<td>( \lnot \text{Cube}(e) \rightarrow \text{Large}(b) \lor \text{Large}(d) )</td>
<td>1</td>
<td>Missing P parens</td>
</tr>
<tr>
<td>5</td>
<td>( \text{Large}(e) \rightarrow \text{Large}(a) )</td>
<td>( e \rightarrow \text{Large}(a) )</td>
<td>2</td>
<td>Elided Predicate</td>
</tr>
<tr>
<td>6</td>
<td>( \text{Tet}(a) \rightarrow \text{FrontOf}(a,d) )</td>
<td>( \text{Tet}(a) \rightarrow \text{InFrontOf}(a,d) )</td>
<td>3i</td>
<td>Incorrect Predicate</td>
</tr>
<tr>
<td>7</td>
<td>( \text{Tet}(a) \rightarrow \text{FrontOf}(a,d) )</td>
<td>( \text{Tet}(a) \rightarrow \text{FrontOf}(a,b) )</td>
<td>3ii</td>
<td>Incorrect Constant</td>
</tr>
<tr>
<td>8</td>
<td>( \text{Tet}(a) \rightarrow \text{FrontOf}(a,d) )</td>
<td>( \text{Tet}(a) \rightarrow \text{FrontOf}(d) )</td>
<td>3ii</td>
<td>Arity Error</td>
</tr>
</tbody>
</table>
## Error Frequencies

<table>
<thead>
<tr>
<th>Error Type</th>
<th>Count</th>
<th>% of All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antecedent–Consequent Reversal</td>
<td>25084</td>
<td>25.86%</td>
</tr>
<tr>
<td>Biconditional for Conditional</td>
<td>17518</td>
<td>18.06%</td>
</tr>
<tr>
<td>Conditional for Biconditional</td>
<td>11362</td>
<td>11.71%</td>
</tr>
<tr>
<td>Negation Error</td>
<td>8954</td>
<td>9.23%</td>
</tr>
<tr>
<td>Incorrect Scope</td>
<td>5422</td>
<td>5.59%</td>
</tr>
<tr>
<td>Failure to Scope</td>
<td>4701</td>
<td>4.85%</td>
</tr>
<tr>
<td>Argument Error</td>
<td>4474</td>
<td>4.61%</td>
</tr>
<tr>
<td>Conjunction for Conditional</td>
<td>3187</td>
<td>3.29%</td>
</tr>
<tr>
<td>Conditional for Conjunction</td>
<td>2091</td>
<td>2.16%</td>
</tr>
<tr>
<td>Biconditional for Conjunction</td>
<td>1514</td>
<td>1.56%</td>
</tr>
</tbody>
</table>
## BiCondForCond Errors

<table>
<thead>
<tr>
<th>Frequency</th>
<th>Percentage</th>
<th>Surface Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>13214</td>
<td>75.43%</td>
<td>S only if S.</td>
</tr>
<tr>
<td>1777</td>
<td>10.14%</td>
<td>S unless S.</td>
</tr>
<tr>
<td>1146</td>
<td>6.54%</td>
<td>S provided S.</td>
</tr>
<tr>
<td>725</td>
<td>4.14%</td>
<td>S if S.</td>
</tr>
<tr>
<td>367</td>
<td>2.09%</td>
<td>If S then if S then S.</td>
</tr>
<tr>
<td>289</td>
<td>1.65%</td>
<td>If S then S.</td>
</tr>
</tbody>
</table>
Agenda

• Approaches to Generation, Past and Present
• The OpenProof Project
• Paraphrase Selection
• A Look at Some Real Data
• Next Steps
Logic to NL Correspondences

\[
\begin{align*}
\text{SEM} & \quad \text{CONNECTIVE} \quad \text{conditional} \\
& \quad \text{ANTECEDENT} \quad A \\
& \quad \text{CONSEQUENT} \quad B \\
\text{CONDITIONAL-FORM} & \quad \text{if-then} \\
\text{CAT} & \quad s \\
\text{DTR1} & \quad \text{LEX} \quad if \\
\text{DTR2} & \quad \text{CAT} \quad s \\
\text{DTR3} & \quad \text{SEM} \quad A \\
\text{DTR4} & \quad \text{LEX} \quad then \\
& \quad \text{CAT} \quad s \\
& \quad \text{SEM} \quad B
\end{align*}
\]
Realisation Classes: Different Realisations of the Conditional
Realisation Classes: Surface Form Effects
Generation Strategy

• Malrules detect the types of errors found in the student's solution
• Each malrule results in directives for the generator to select structures that have particular features
• In complex cases there may be conflicting requirements
  – The generator should try to select the combination of features most likely to result in understanding
  – Best choice determined by weightings derived from the user and task model
Next Steps

• Further development of the error taxonomy and malrules
• Characterisation of a range of paraphrase rules to deal with the common cases
• Implementation of a prototype generator
Conclusions

• Traditional NLG requires:
  – a rich semantic input representation to motivate linguistic distinctions
  – widely varying contexts of use to motivate variation in output
• OpenProof + an immense student base provides both
• Other possibilities for the same approach:
  – Tailored advice in language learning
  – Customised web pages based on browsing history