The Generation of Referring Expressions: Past, Present and Future

Robert Dale
Robert.Dale@mq.edu.au
The Aims of This Talk

- To provide a brief history of work in referring expression generation
- To identify where there’s still work to do
- To suggest an alternative framework
Outline

• A Definition of the Problem
• What We’ve Achieved in 25 Years
• Where We’ve Fallen Short
• Where We Go From Here
The Context

• NLG as part of Good Old Fashioned Natural Language Processing
  – Graph-structured knowledge representation
  – Entities and events have symbolic identifiers
• The role of NLU: to build such a representation
• The role of NLG: to describe parts of such a representation
The Context

<table>
<thead>
<tr>
<th>Focus on text mapping</th>
<th>For example, Textual Entailment</th>
<th>Referring Expression Generation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Focus on abstract representations</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Graph-Structured Knowledge Representations

Input Texts

Natural Language Analysis

Natural Language Generation

Output Text
Entities, Events, Attributes and Values

- **x1**: HAS-NAME
  - JOHN

- **AGENT-OF**: FRIEND-OF
  - **e1**: GIVING
    - BENEFICIARY
    - INSTANCE-OF
      - **x2**: NAME
        - FRED
      - **x3**: OBJECT
        - HAS-COLOR
          - **JUMPER**: HAS-COLOR
            - RED
Natural Language Generation

Document Planning

Micro Planning

Surface Realization

Content Determination
Text Structuring

Lexicalisation

Aggregation

Referring Expression Generation

Syntax, morphology, orthography and prosody
Referring Expression Generation

Input propositions:
owns(j, j1), wears(j, j1, s)

NP semantics:
isa(j1, jumper) \land colour(j1, red)
Referring Expression Generation

Internal Symbol → Process → NP
Referring Expression Generation

\[ x_1 \xrightarrow{\text{Process}} \text{NP} \xrightarrow{\text{the red jumper}} \]
Referring Expression Generation

\[ x_1 \rightarrow \text{What} \rightarrow \text{How} \rightarrow \text{NP} \]

the red jumper
Referring Expression Generation

What

[type: jumper
colour: red]

How

NP

the red jumper
Referring Expression Generation

\[ \text{x1} \rightarrow \text{What} \rightarrow \left[ \text{type: jumper} \right. \left. \text{colour: red} \right] \rightarrow \text{How} \rightarrow \text{NP} \rightarrow \text{the red jumper} \rightarrow \text{NP} \rightarrow \text{the jumper which is red} \]
Generating Referring Expressions

The standard definition of the problem:

• given an intended referent;
• given a knowledge base of entities characterised by attribute value pairs; and
• given a context consisting of other entities that are salient;

Then:

• choose a set of attribute value pairs that uniquely identify the intended referent
Outline

• A Definition of the Problem
• What We’ve Achieved in 25 Years
• Where We’ve Fallen Short
• Where We Go From Here
What We’ve Achieved in the Last 25 Years: Starting Points, 1980-1985

• Dave McDonald’s thesis [1980]
  – A complete model of NLG, aiming at incrementality and psycholinguistic plausibility
  – introduced distinction between initial and subsequent reference as distinct NLG problems

• Doug Appelt’s thesis [1982]
  – Planning natural language utterances that satisfy several goals simultaneously
  – introduced idea of NPs performing both reference and other functions

• Kathy McKeown’s thesis [1982]
  – Describing database objects, with a focus on discourse coherence
  – introduced interaction between content choice and pronominalisation
What We’ve Achieved in the Last 20 Years: A Focus on Algorithms

The standard framework:

- Given an intended referent and a context of potential distractors, construct a distinguishing description that uniquely identifies the intended referent for the hearer
A Skeletal Algorithm

Given an intended referent x:

begin

if x is in focus then use a pronoun

elseif x has been mentioned already

then build a definite noun phrase

else build an initial indefinite reference

end
Computing Distinguishing Descriptions

Initial Conditions:
\[ C_r = \langle \text{all entities} \rangle; \quad P_r = \langle \text{all properties true of } r \rangle; \quad L_r = \{\} \]

1 Check Success
   if \(|C_r| = 1\) then return \(L_r\) as a distinguishing description
   elseif \(P_r = 0\) then return \(L_r\) as a non-dd
   else goto Step 2.

2 Choose Property
   for each \(p_i \in P_r\) do: \(C_{r_i} \leftarrow C_r \cap \{x \mid p_i(x)\}\)
   Chosen property is \(p_j\), where \(C_{r_j}\) is smallest set.
   goto Step 3.

3 Extend Description (wrt the chosen \(p_j\))
   \[ L_r \leftarrow L_r \cup \{p_j\}; \quad C_r \leftarrow C_{r_j}; \quad P_r \leftarrow P_r \setminus \{p_j\}; \quad \text{goto Step 1.}\]
Advances Over the Last 20 Years: Algorithm Development, 1985-1995

• Appelt 1985: pragmatic aspects of referring expression generation
• Reithinger 1987: integration of linguistic reference and pointing
• Dale 1989: distinguishing descriptions incorporating one-place predicates only
• Reiter 1990: computational complexity problems
• Dale and Haddock 1991: constraint-based extension to handle relational properties
• Dale and Reiter 1992: incremental algorithm for one-place predicates, more in line with psycholinguistic data
Advances Over the Last 20 Years: More Algorithm Development, 1995-2000

- Creaney 1996: generating quantifiers
- Horacek 1996: integration of constraint-based and incremental approaches
- Horacek 1997: incorporation of linguistic constraints to ensure expressibility
- Bateman 1999: an aggregation-based metaphor for referring expression generation
- Stone and Webber 1998: simultaneous semantic and syntactic construction
Advances Over the Last 20 Years: Yet More Algorithm Development, 2000-2005

• Stone 2000: referring to sets
• Krahmer et al 2001, 2002: reconceptualisation as a subgraph construction problem
• Krahmer and Theune 2002: incorporation of a treatment of salience
• Van Deemter 2002: extension of the incremental algorithm to handle negation and disjunction of properties
• Siddharthan and Copestake 2002: application to text simplification
• Gardent and Striegnitz 2003: extension to bridging descriptions
• Kibble and Power 2004: using centering in reference generation
Outcomes

• A number of base algorithms within the standard framework
• Extensions to accommodate sets, negation, disjunction, bridging reference, salience, pointing, linguistic constraints, quantifiers
• Some explorations into other ways of thinking about the problem
Outline

- A Definition of the Problem
- What We’ve Achieved in 25 Years
- Where We’ve Fallen Short
- Where We Go From Here
Where We’ve Fallen Short
#1: Generating Pronouns

Given an intended referent $x$:

begin

if $x$ is in focus

then use a pronoun

elseif $x$ has been mentioned already

then build a definite noun phrase

else build an initial indefinite reference

end
Where We’ve Fallen Short
#2: Generating Initial References

Given an intended referent $x$:

begin

if $x$ is in focus

then use a pronoun

elseif $x$ has been mentioned already

then build a definite noun phrase

else build an initial indefinite reference

end
What We’ve Not Achieved in the Last 30+ Years

• Terry Winograd’s SHRDLU [1971, 1972]
  – By "it", I assume you mean the block which is taller than the one I am holding.

• Anthony Davey’s Tic-tac-toe program [1973, 1978]
  – If you had blocked my line, you would have threatened me, but you took the corner adjacent to the one which you took first and so I won by completing my line.
Where We’ve Fallen Short

Conclusion:

• There’s a well-developed existing framework for work in the area

• There are a number of important phenomena not yet properly addressed

• So, there’s a healthy research agenda waiting for eager PhD students
The Drunk Under The Lamp Post?

Important Problems
A Dark Corner

• How does the generation of *one*-anaphora fit into the standard conception of the problem of referring expression generation?
One-Anaphora

Contrasting individuals:
• John owns the red jumper.
• Robert owns the blue one.

Introducing a representative sample of a set:
• John has several jumpers.
• The warmest one is made from Shetland wool.

Referring to a new specimen of an introduced type:
• John has several old jumpers.
• Mary wants to buy him a new one.
The Top-Level Algorithm

Given an intended referent $x$:

begin

if $x$ is in focus
then use a pronoun
elseif $x$ has been mentioned already
then build a definite noun phrase
else build an initial indefinite reference

end
Generating One-Anaphora: Compare Syntactic Structures

\[ \text{Generate } x_1 \rightarrow \text{the red jumper} \]
\[ \text{Generate } x_2 \rightarrow \text{the green one} \]

Diagram:

- Generate \( x_1 \)
- Generate \( x_2 \)
- Compare

Trees:

- \( \text{NP} \)
  - \( \text{the red jumper} \)
- \( \text{NP} \)
  - \( \text{the green jumper} \)
Generating One-Anaphora: Compare Semantic Structures

Better Solution:

• Generate semantic form that distinguishes the head of the expression and compare with preceding context
  - \( \text{type}(x_1, \text{jumper}) \land \text{colour}(x_1, \text{red}) \)
  - \( \text{type}(x_2, \text{jumper}) \land \text{colour}(x_2, \text{blue}) \)

• *One*-anaphora is possible provided the same type is shared, along with zero or more other properties
Generating One-Anaphora: Compare Semantic Structures

\[ x_1 \rightarrow \text{What} \rightarrow \left[\begin{array}{l}
\text{type: jumper} \\
\text{colour: red}
\end{array}\right] \rightarrow \text{How} \rightarrow \text{the red jumper} \]

\[ x_2 \rightarrow \text{What} \rightarrow \left[\begin{array}{l}
\text{type: jumper} \\
\text{colour: blue}
\end{array}\right] \rightarrow \text{Compare} \rightarrow \text{How} \rightarrow \text{the blue one} \]
Generating One-Anaphora

Still not ideal:

- Requires building entire semantic structure – and choosing all the properties to use – before we know if one-anaphora is possible
Where Do One-Anaphora Processes Belong?

Internal Symbol → What → Semantic Content → How → NP

Compare

Compare

Compare
Uses of One-Anaphora: Case #1

1a John has a red t-shirt.
1b Bill has a blue one.

• Speaker contrasts two pieces of information.
Uses of One-Anaphora: Case #2

2a John has a red coat and a blue one.
2b He wears the red one on Sundays and the blue one on Mondays.
• Speaker decides to comment on two similar entities.
Uses of One-Anaphora: Case #3

3a Bill has two red t-shirts.
3b He wears the smaller one on Mondays.

• Speaker introduces a set of similar entities then elaborates on an element of that set.
The Functions of One-Anaphora

Amongst others:
• introduce a referent of the same type as one previously introduced and contrast it
• select an element from a set of already introduced entities

Key point:
• these are not arbitrary sequences of sentences---they are “spoken as pairs”

Proposal:
• control referring expression generation by the discourse function’s requirements
How To Do This: Lexicalised Discourse Schemata

Contrast Goal + Appropriate Configuration of Properties in KB

Discourse Pattern

Contrast

\[ x_1 \quad x_2 \]
Outline

• A Definition of the Problem
• What We’ve Achieved in 25 Years
• Where We’ve Fallen Short
• Where We Go From Here
An Alternative to the Standard Framework

- Referring expressions are not generated in a vacuum
- Recognize that reference is a discourse phenomenon, selected for at an early stage in the generation process
Some Discourse Functions of Reference

- For one-anaphora:
  - contrast-with-other-entity
  - select-element-from-set
- For pronominal reference:
  - maintain-as-focus
  - shift-into-focus
- For initial reference:
  - introduce-entity
- For subsequent reference:
  - distinguish-entity-from-distractors
  - attribute-additional-information
Conclusions

• The standard architecture adopted in NLG systems assumes the single discourse purpose of reference is to distinguish an entity from others in the context

• A richer model of reference needs to take account of other communicative functions
Conclusions

• May also suggest a different strategy for NL understanding: don't just work out what the referent of an expression is, also work out the discourse purpose in introducing the reference

• Potentially important for
  – Text summarisation
  – Question answering
  – Information extraction
The Moral

• It’s good to have a standard framework that unifies the field, but …
• … stay sober
• … make sure you shine your torch into unexplored corners