AN INTRODUCTION TO CONTENT DETERMINATION

Gerard Casamayor
Chris Mellish
Contents

1. The place of Content Determination
2. Styles of Content Determination
3. Methods for Content Determination
4. Examples
5. Content Determination from SW data
1. The place of Content Determination
Content Determination

• The main interface between the NLG system and the domain/application/outside world.

• Decides “what to say” in terms of domain concepts
Reiter and Dale (2000) NLG pipeline

The NLG System

Document Planning

- Content Determination
- Ordering and Structuring

Sentence planning

Surface Realisation

The application

“what to say?”

“how to say it?”

Domain dependent

Language dependent

“what to say?”

“how to say it?”

Domain dependent

Language dependent
NLG pipelines in dialogue systems

White, Clark, and Moore (2010)

Walker, Rambow, and Rogati (2002)

- See slides on Statistical Natural Language Generation by M. White (2010)
Why is Content Determination hard?

a) Hard to develop reusable approaches:
   • Multiple domains
   • Multiple input data formats

Continuous signal, e.g. BabyTalk (Portet et al. 2007)

Semantic data (Bouttaz et al. 2011)

Tabular (Angeli et al. 2010)
Why is Content Determination hard?

a) It may not naturally provide enough information to satisfy what the language needs, or it may not produce something that can be elegantly expressed – the “generation gap” (Meteer 92), e.g.

• How much material can be put into a single sentence/ paragraph/ tweet/ A4 page?

• Is it easy to express “pleasure in another person’s misfortune” (yes, if you are speaking German)?
Why is Content Determination hard?

c) It may not be able to choose among alternatives which are equivalent in the application but which make a big difference in the language, e.g. the “problem of logical form equivalence” (Shieber 93):

<table>
<thead>
<tr>
<th>(Item)</th>
<th>String</th>
<th>Canonical Logical Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>John threw a large red ball.</td>
<td>$\exists x.\text{throw}(j, x) \land \text{large}(x) \land \text{red}(x) \land \text{ball}(x)$</td>
</tr>
<tr>
<td>(ii)</td>
<td>John threw a red ball that is large.</td>
<td>$\exists x.\text{throw}(j, x) \land \text{red}(x) \land \text{ball}(x) \land \text{large}(x)$</td>
</tr>
<tr>
<td>(iii)</td>
<td>John threw a large ball that is red.</td>
<td>$\exists x.\text{throw}(j, x) \land \text{large}(x) \land \text{ball}(x) \land \text{red}(x)$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(Item)</th>
<th>String</th>
<th>Canonical Logical Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>(iv)</td>
<td>Clapton was the leader of Derek and the Dominos.</td>
<td>$\text{the}(x, \text{leader-of}(dd, x), c = x)$</td>
</tr>
<tr>
<td>(v)</td>
<td>The leader of Derek and the Dominos was Clapton.</td>
<td>$\text{the}(x, \text{leader-of}(dd, x), x = c)$</td>
</tr>
<tr>
<td>(vi)</td>
<td>Clapton led Derek and the Dominos.</td>
<td>$\text{led}(c, dd)$</td>
</tr>
</tbody>
</table>
2. Styles of Content Determination
Top-down vs Bottom-up

- Top-down (goal driven, backwards) processing looks at how to find content to support one of a known set of possible text types:
  - Satisfy communicative goals
  - Good when there are strong conventions for what texts should be like
  - Making sure the text will have a coherent structure

- Bottom-up (data driven, forwards) processing looks at what the application makes available and seeing how a text can be made from it:
  - Diffuse goals
  - Working out what is most important/interesting
  - Good when the form of the text needs to vary a lot according to what is actually there
Separate task vs interleaved

- Reiter and Dale’s pipeline shows Content Determination as a separate module.
- But there are dependencies between CD and other NLG tasks.
  - Error propagation: the generation gap may become evident during surface realization.
  - Alternative architectures attempt to capture interdependencies:
    - NLG systems as a unified planning problem, e.g. (Hovy 1993), (Young and Moore 1994)
    - Cascade of classifiers in the Discrete Optimization Model of (Marciniak, Strube 2005)
    - Hierarchical Reinforcement Learning for Adaptive Text Generation (Dethlefs et al. 2010)
Types of input data

• Many types of input data
• Input contents may require interpretation:

1. Continuous data signal or raw numerical data requires assessment
   • E.g. infer qualitative rating *Strong* from quantitative wind speed readings SUMTIME (Sripada et al. 2003)

2. Some aspects of the input data not explicitly encoded but inferable.
   • E.g. football match score is 1-1. Infer this result is a draw. (Bouayad-Agha et al. 2011)

3. What are the units to be selected? What is the granularity of content determination?
   • Message determination
     • In relation databases: a single cell, a whole row, a subset of the row?
     • In Semantic Web datasets: a triple, all triples about an individual?
Context

- Content determination may take into account some of the following:
  - Targeted genre: term definition, report, commentary, narrative, etc.
  - Targeted audience: lay person, informed user, domain expert, etc.
  - Request: information solicitation, decision support request, etc.
  - Communicative goal: exhaustive information on a theme, advice, persuasion, etc.
  - User profile: user preferences, needs or interests in the topic, individual expertise, previous knowledge, discourse history, etc.
3. Methods for Content Determination
Templates and schemas

• Simple and effective way of capturing observed regularities in target texts

• Templates lack flexibility

• Schemas make up for that by introducing expansion slots to be completed with contents or linguistic information.
  • (McKeown 1992)
  • MIAKT and ONTOSUM systems, (Bontcheva and Wilks 2004), (Bontcheva 2005)

• Templates and schemas can be used to by-pass NLG altogether
Automated planning

• Find sequence of actions to satisfy a goal

• Knowledge about domain and how to communicate it is modeled using planning languages (STRIPS, ADL, PDDL).

• The planning problem is addressed using a general problem solver, e.g. hierarchical planning with goal decomposition.
  • (Hovy 1993), (Young and Moore 1994), (Carenini and Moore 2006), (Paris et al. 2010).

• Content determination and structuring (and even other NLG tasks!) are handled together.
  • Planning guided by rhetorical operators that ensure coherence of text
Automated reasoning

- Start from a Knowledge Base (KB) encoding knowledge about the domain.
  - Rich semantics: knowledge representation languages, ontologies
  - Not created specifically for NLG purposes

- Types of knowledge (Rambow 1990):
  - Domain knowledge: input data, its syntax and semantics
  - Communication knowledge: domain-independent knowledge about language, discourse, etc.
  - Domain communication knowledge: how to communicate domain data

- Reasoning requires explicit, symbolic representations of how to communicate data (rules, ontologies, etc.) or a special type of inference suitable for NLG.
  - Donnell et al. (2001), (Bouayad-Agha et al. 2011, 2012), (Bouttaz 2011)
  - (Mellish and Pan 2008)
Graph-based methods

• Build a graph representation of the input data and operate on this representation.

• Graph may be reflect semantic relations between data but also statistical information, e.g. using weights.

• Two mechanisms:
  1. Explore the graph from a central point, e.g. entity of interest.
     • In Donnell et al. (2001) and Dannélls et al. (2009), a rooted content graph is navigated in search of relevant data.
  2. Apply a global graph algorithm to weight all nodes/Edges and find most relevant subset.
     • In Demir et al. (2010) PageRank is applied to find a subset of the content graph that maximizes relevance and reduces redundancies.
Statistical methods

• General statistical approach:
  1. Construct a general model that assigns probabilities to outputs, given inputs
  2. Provide training data to the model, in order to tune the internal parameters
  3. Present the trained model with a real input
  4. Search for the output which maximises the probability according to the model

• Model can be trained from corpora of human authored texts aligned with contents.
  a. Manual annotation
  b. Automatic linkage of texts and contents
## Statistical methods

<table>
<thead>
<tr>
<th>System</th>
<th>Model</th>
<th>Input</th>
<th>Search strategy</th>
<th>Training data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Barzilay and Lapata 2005</td>
<td>Weighted graph + multiple classifiers</td>
<td>Database rows</td>
<td>Minimal cut partition</td>
<td>Automatically aligned corpus</td>
</tr>
<tr>
<td>Kelly et al. 2009</td>
<td>Single classifier</td>
<td>Semistructured data</td>
<td>None</td>
<td>Automatically aligned corpus</td>
</tr>
<tr>
<td>Belz 2008</td>
<td>PCFG with estimated weights</td>
<td>Tabular</td>
<td>Greedy</td>
<td>Manually annotated corpus</td>
</tr>
<tr>
<td>Konstas et al. 2013</td>
<td>PCFG with estimated weights</td>
<td>Database rows</td>
<td>CYK</td>
<td>Manually annotated corpus</td>
</tr>
<tr>
<td>Rieser et al. 2010</td>
<td>Markov Decision Process</td>
<td>Database cells</td>
<td>Reinforcement Learning</td>
<td>Feedback from simulated user</td>
</tr>
<tr>
<td>Dethlefs et al. 2011</td>
<td>Markov Decision Process</td>
<td>Simulated data</td>
<td>Hierarchical reinforcement learning</td>
<td>Feedback from simulated user</td>
</tr>
</tbody>
</table>
Wrapping up

• Styles:
  1. Top-down vs bottom-up
  2. Separate task vs interleaved
  3. Type of input data
  4. Context

• Methods:
  1. Templates and schemas
  2. Automated planning
  3. Automated reasoning
  4. Graph-based methods
  5. Statistical methods

• Methods aren’t mutually exclusive, they can be combined in the same implementation.
4. Examples
Example 1: Paris et al. 2010

• Input data:
  • Knowledge Base with explicit semantics (domain ontology).
  • Granularity: coarse-grained units of information.

• Top-down, goal-driven.

• Interleaved with structuring

• Context: user profile, user history, explicit communicative goals

• Methods: hierarchical planning.
Example 1: Paris et al. 2010

- Text planning module produces discourse structures where information is connected with rhetorical relations.
  - Discourse trees from Rhetorical Structure Theory (RST).

- The system maintains a library of plans capable of producing such trees top-down from an initial communicative goal.
  1. The plans hierarchically decompose goals
  2. Recursive application of plans until all goals are satisfied and a discourse structure is produced
Example 1: Paris et al. 2010

- Plans access knowledge base and user model and history so that their effects are conditioned by domain knowledge, available data and user preferences.

- Discourse plans decide what content should be included, i.e. they perform content determination too.
  - Focus on coarse-grained units of information.

```xml
<operator>
  <id>inform-with-title</id>
  <description>
    Posts an inform goal and includes a title for it
  </description>
  <effect>(inform-with-title ?user text ?data ?title)</effect>
  <nucleus>
    <value>(inform ?user text ?data)</value>
  </nucleus>
  <satellite>
    <type>optional</type>
    <relation>Preparation</relation>
    <value>(Inform user text ?title)</value>
  </satellite>
</operator>
```
Example 2: Barzilay and Lapata 2005

- Input data:
  - structured -> relational database containing events in football matches. Size: 73,400 rows distributed across 17 tables.
  - Granularity: a row of a table in the DB

- Bottom-up

- Separate task

- Methods: statistics, graph algorithm.
Example 2: Barzilay and Lapata 2005

<table>
<thead>
<tr>
<th>Passing</th>
<th></th>
<th></th>
<th>AVG</th>
<th>TD</th>
<th>INT</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLAYER</td>
<td>CP/AT</td>
<td>YDS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Brunell</td>
<td>17/38</td>
<td>192</td>
<td>6.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Garcia</td>
<td>14/21</td>
<td>195</td>
<td>9.3</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rushing</th>
<th></th>
<th></th>
<th>AVG</th>
<th>LG</th>
<th>TD</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLAYER</td>
<td>REC</td>
<td>YDS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Suggs</td>
<td>22</td>
<td>82</td>
<td>3.7</td>
<td>25</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fumbles</th>
<th></th>
<th></th>
<th>REC</th>
<th>YDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLAYER</td>
<td>FUM</td>
<td>LOST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coles</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Portis</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Davis</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Little</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Suggs rushed for 82 yards and scored a touchdown in the fourth quarter, leading the Browns to a 17-13 win over the Washington Redskins on Sunday. Jeff Garcia went 14-of-21 for 195 yards and a TD for the Browns, who didn’t secure the win until Coles fumbled with 2:08 left. The Redskins (1-3) can pin their third straight loss on going just 1-for-11 on third downs, mental mistakes and a costly fumble by Clinton Portis. Brunell finished 17-of-38 for 192 yards, but was unable to get into any rhythm because Cleveland’s defense shut down Portis. The Browns faked a field goal, but holder Derrick Frost was stopped short of a first down. Brunell then completed a 13-yard pass to Coles, who fumbled as he was being taken down and Browns safety Earl Little recovered.
Example 2: Barzilay and Lapata 2005

- Database rows are automatically aligned to sentences in a corpus of match reports using simple anchor-based techniques.
- Overlap between cell values in row and tokens in sentence.
- This works because proper names and numbers which are easy to align occur with high frequency.

<table>
<thead>
<tr>
<th>Entity Type</th>
<th>Attr</th>
<th>Inst</th>
<th>%Aligned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defense</td>
<td>8</td>
<td>14,077</td>
<td>0.00</td>
</tr>
<tr>
<td>Drive</td>
<td>10</td>
<td>11,111</td>
<td>0.00</td>
</tr>
<tr>
<td>Play-by-Play</td>
<td>8</td>
<td>83,704</td>
<td>3.03</td>
</tr>
<tr>
<td>Fumbles</td>
<td>8</td>
<td>2,937</td>
<td>17.78</td>
</tr>
<tr>
<td>Game</td>
<td>6</td>
<td>469</td>
<td>0.00</td>
</tr>
<tr>
<td>Interceptions</td>
<td>6</td>
<td>894</td>
<td>45.05</td>
</tr>
<tr>
<td>Kicking</td>
<td>8</td>
<td>943</td>
<td>26.93</td>
</tr>
<tr>
<td>Kickoff-returns</td>
<td>8</td>
<td>1,560</td>
<td>5.24</td>
</tr>
<tr>
<td>Officials</td>
<td>8</td>
<td>464</td>
<td>0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Entity Type</th>
<th>Attr</th>
<th>Inst</th>
<th>%Aligned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Passing</td>
<td>5</td>
<td>1,185</td>
<td>59.90</td>
</tr>
<tr>
<td>Team comparison</td>
<td>4</td>
<td>14,539</td>
<td>0.00</td>
</tr>
<tr>
<td>Punt-returns</td>
<td>8</td>
<td>940</td>
<td>5.74</td>
</tr>
<tr>
<td>Punting</td>
<td>9</td>
<td>950</td>
<td>0.87</td>
</tr>
<tr>
<td>Receiving</td>
<td>8</td>
<td>6,337</td>
<td>11.19</td>
</tr>
<tr>
<td>Rushing</td>
<td>8</td>
<td>3,631</td>
<td>9.17</td>
</tr>
<tr>
<td>Scoring-sum</td>
<td>9</td>
<td>3,639</td>
<td>53.34</td>
</tr>
<tr>
<td>Team</td>
<td>3</td>
<td>4</td>
<td>0.00</td>
</tr>
</tbody>
</table>
Example 2: Barzilay and Lapata 2005

- Links between a row and a sentence constitute positive examples of selection of the row.

- Collective selection of database rows belonging to a match:
  1. A graph is built where nodes are rows in the database and edges indicate semantic relatedness, i.e. the connected rows share at least one attribute value.
  2. Nodes weights are predictions of a set of models trained using machine learning.
  3. Edge pruning: Discard edges where both rows have different selection distribution across documents.
  4. Edge weighting: use simulated annealing to obtain a global assignment of weights according to node weights and edges.

- Result: weighted constraints between pairs of rows belonging to a match.
Example 3: Konstas and Lapata 2013

• Input data:
  • structured -> relational databases belonging to three domains: Robocup game finals, weather forecasts and air travel.
  • Granularity: a row of a table in the DB

• Data-driven

• Content determination interleaved with ordering and surface realization.

• Methods: statistics, symbolic grammar (PCFG).
Figure 1: Input-output examples for (a) query generation in the air travel domain, (b) weather forecast generation, and (c) sportscasting.
Example 3: Konstas and Lapata 2013

- Probabilistic Context Free Grammar (PCFG) as set of rewrite rules: rewrite structure of DB (rows, cells) into words.
  - The weights of the PCFG are estimated by applying the CYK parser and the grammar to texts in a corpus of verbalizations of the DB.
  - For each text a hypergraph is built and the weights are updated.
  - Generation maximizes the PCFG grammar and an n-gram language model.
  - Uses datasets of DB records (manually) paired with texts verbalizing them.

\[
Gcs \quad \begin{align*}
1. & S \rightarrow R(\text{start}) \\
2. & R(ri.t) \rightarrow FS(rj,start)R(\text{rj}.t) \\
3. & R(ri.t) \rightarrow FS(rj,start) \\
4. & FS(r,r.f_i) \rightarrow F(r,r.f_i)FS(r,r.f_i) \\
5. & FS(r,r.f_i) \rightarrow F(r,r.f_i) \\
6. & F(r,r.f) \rightarrow W(r,r.f)F(r,r.f) \\
7. & F(r,r.f) \rightarrow W(r,r.f) \\
\end{align*}
\]

\[
Gsurf \quad \begin{align*}
8. & W(r,r.f) \rightarrow \alpha \\
9. & W(r,r.f) \rightarrow \text{gen}(f,v) \\
\end{align*}
\]
Example 3: Konstas and Lapata 2013

Cloudy, with temperatures between 10 and 20 degrees.
4. Content Determination from SW Data
Why does it matter?

1. A common interface for accessing and reasoning about data:
   - HTTP, URIs, RDF, RDFS, OWL, SPARQL, reasoners, etc.

2. Large (and growing) amounts of Linked Open Data (LOD) belonging to multiple domains.
   - NLG can be used to make data accessible to humans.


4. NLG-relevant knowledge can be modeled using SW standards and shared LD publishing standards.
   - E.g. Lemon-encoded lexical resources, NIF corpora.

5. Advances in Information Extraction means that corpora of texts annotated with SW data can be created automatically.

6. More research is needed!
Content Determination from SW data

- Four main communicative goals in NLG approaches for SW data:
  i. to say almost all there is to say about some input object (i.e., class, query, constraint, whole graph)
  ii. to verbalize content interactively selected by the user
  iii. To verbalize the most typical facts found in target texts
  iv. To verbalize the most relevant facts according to the context

- No (or trivial) Content Determination in (i) and (ii)
- Templates and schemas used for (iii)
- More elaborate strategies (i.e. statistical methods, graph-based) could be used.
Annotating texts with SW data

- Entity Linking
  - Named entity recognition + disambiguation against a dataset
  - Wikipedia/DBPedia/BabelNet

- Coreference resolution

- Annotation of relations
  - Deep parsers
    - Boxer, Mate tools, Abstract Meaning Representation (AMR) parsers.
  - Semantic Role Labelling
    - FrameNet, VerbNet, PropBank.
Selecting RDF data

• RDF can be viewed as a graph.
  • Properties as edges
  • Entity-sharing as edges
  • Most LD datasets are huge graphs!

• Granularity of selection:
  • Single triple ≠ statement
  • Blank nodes
  • But N-ary relations are expressed with multiple triples

• Statistical methods:
  • Ontologies provide features for the creation of models based on data, e.g. ontological types.
Conclusions

• Content determination is very important (in many applications, the ability of the machine to find relevant material is perhaps more important than how the material is stated)

• Good content determination often involves specialised domain reasoning

• Statistical approaches tend to learn CD as a part of a simple complete NLG pipeline.

• More and more NLG applications are starting from SW data.
References

• Marie Meteer, “Bridging the ‘Generation Gap' between Text Planning and Linguistic Realization” Computational Intelligence. 7(4) Special issue on Natural Language Generation, 1992.
• Rambow, Owen. "Domain communication knowledge." Fifth International Workshop on Natural Language Generation. 1990.