In the beginning there was an Intelligent Query Interface... that needed Natural Language Support. (Part II)

Paolo Dongilli

KRDB Centre
Faculty of Computer Science
Free University of Bozen-Bolzano, Italy

KRDB Seminar – May 24, 2006
Natural Language Generation
- Introduction
- Approaches to Text Realization
- Overview of Linguistic Formalisms

Systemic-Functional Linguistics
- Introduction
- Systemic-Functional Grammar
- An SFG-based Generation System

Query Interface with Natural Language Support
- The New User Interface
- System Internals

Paolo Dongilli
I look for an **off-roader** that runs on **diesel**. It is sold by a **car dealer** which is located in **Germany**, the make is **Land Rover** and the model is **Defender**. The **off-roader** is equipped with **A/C, central locking, and leather seats**.

I need the **price**, and the **mileage** of the **off-roader**, the **name**, the **city**, and the **phone number** of the car dealer.

---

**The New Query Interface**

<table>
<thead>
<tr>
<th>generalize:</th>
<th>car</th>
</tr>
</thead>
<tbody>
<tr>
<td>specialize:</td>
<td></td>
</tr>
<tr>
<td>must be also:</td>
<td>new car, old car</td>
</tr>
<tr>
<td>related concepts:</td>
<td>equipped with <strong>characteristic (ABS, immobilizer, electric windows)</strong>, <strong>exterior color</strong>, runs on fuel (<strong>electrical energy, gasoline, natural gas, propane</strong>), <strong>number of doors</strong>, <strong>power</strong>, <strong>registration date</strong>, <strong>transmission (automatic transmission, manual gear transmission)</strong></td>
</tr>
</tbody>
</table>

...that needed Natural Language Support.
What’s Natural Language Generation?

- A subfield of Artificial Intelligence and Computational Linguistics concerned with building computer software systems that can produce meaningful texts in one or more human languages from some underlying nonlinguistic representation of information.

- NLG Systems use knowledge about language and the application domain to automatically produce documents, reports, help messages and other kinds of text.
What’s Natural Language Generation?

- A subfield of Artificial Intelligence and Computational Linguistics concerned with building computer software systems that can produce meaningful texts in one or more human languages from some underlying nonlinguistic representation of information.

- NLG Systems use knowledge about language and the application domain to automatically produce documents, reports, help messages and other kinds of text.
Tasks of Natural Language Generation

- **Text Planning**
  - Content determination
  - Discourse structuring (e.g. RST [Mann and Thompson, 1988])

  ➞ **Text Plan**

- **Sentence Planning**
  - Lexicalization
  - Aggregation
  - Referring expression generation

  ➞ **Sentence Plan**

- **Text Realization**

  ➞ **Sequence of words**

Paolo Dongilli... that needed Natural Language Support.
Tasks of Natural Language Generation

- Text Planning
  - Content determination
  - Discourse structuring (e.g. RST [Mann and Thompson, 1988])

\[\Rightarrow\] Text Plan

- Sentence Planning
  - Lexicalization
  - Aggregation
  - Referring expression generation

\[\Rightarrow\] Sentence Plan

- Text Realization
  \[\Rightarrow\] Sequence of words

Paolo Dongilli... that needed Natural Language Support.
Tasks of Natural Language Generation

- **Text Planning**
  - Content determination
  - Discourse structuring (e.g. RST [Mann and Thompson, 1988])

  \[\rightarrow\] **Text Plan**

- **Sentence Planning**
  - Lexicalization
  - Aggregation
  - Referring expression generation

  \[\rightarrow\] **Sentence Plan**

- **Text Realization**

  \[\rightarrow\] **Sequence of words**
Approaches to Text Realization

Four basic approaches proposed by [Hovy, 1997]:

1. Canned Text
2. Template-Based Realization
3. Phrase-Based Realization
4. Feature-Based Realization
Four basic approaches proposed by [Hovy, 1997]:

1. Canned Text
2. Template-Based Realization
3. Phrase-Based Realization
4. Feature-Based Realization
Four basic approaches proposed by [Hovy, 1997]:

1. Canned Text
2. Template-Based Realization
3. Phrase-Based Realization
4. Feature-Based Realization
Four basic approaches proposed by [Hovy, 1997]:

1. Canned Text
2. Template-Based Realization
3. Phrase-Based Realization
4. Feature-Based Realization
Four basic approaches proposed by [Hovy, 1997]:

1. Canned Text
2. Template-Based Realization
3. Phrase-Based Realization
4. Feature-Based Realization
Linguistic Formalisms for Realization

- Systemic Functional Grammar (SFG)
- Meaning-Text-Theory (MTT)
- Functional Unification Grammar (FUG), progenitor of the Functional Unification Formalism (FUF)
- Tree Adjoining Grammars (TAG)
- Categorial Unification Grammar
- Lexical Functional Grammar
- Government and Binding Theory
- Head-driven Phrase Structure Grammar (HPSG)
Origins of SFL

- Systemic-Functional Linguistics (SFL) is a major linguistics theory.
Origins of SFL (Malinowski)

- Rooted in anthropology (Bronislaw Kasper Malinowski, 1935)
- Important contributions to early modern linguistics from an anthropological perspective: meaning as function in context

Context

- Verbal Context
- Non-Verbal Context

- Context of Situation
- Context of Culture

Paolo Dongilli...that needed Natural Language Support.
Earliest formulation of SFL as linguistic theory dates back to John Rupert Firth (1957), founder of modern British linguistics and the first Professor of General Linguistics in the UK.

Firth disagreed with the American structuralists of his time (led by Bloomfield), because they were concerned only with the “anatomy” of language.
Origins of SFL (Firth)

- need for linguistics to give equal importance to both the “anatomy” and “physiology”

<table>
<thead>
<tr>
<th>anatomy</th>
<th>physiology</th>
</tr>
</thead>
<tbody>
<tr>
<td>chain</td>
<td>choice</td>
</tr>
<tr>
<td>syntagmatic</td>
<td>paradigmatic</td>
</tr>
<tr>
<td>structural</td>
<td>systemic</td>
</tr>
<tr>
<td>formal</td>
<td>functional</td>
</tr>
<tr>
<td>logical</td>
<td>rhetorical</td>
</tr>
</tbody>
</table>
Origins of SFL (Halliday)

- Further developed by Michael A. K. Halliday since 1961
- Michael Halliday, Firth’s pupil and successor at London, disagreed with the American formalists (led by Chomsky).
- Birth of “Neo-Firthian linguistics”, or the “London school of linguistics”
- Halliday developed a systematic and comprehensive theory of language, with a new terminology of its own. This theory, expounded in Halliday’s many publications, became known as Systemic Functional Grammar.
Why systemic? Why functional?

- **systemic** because of his development of detailed system networks for many areas of the English grammar and of other languages’ grammars.
- **functional** because of his development of the theory of the ideational, interpersonal and textual metafunctions.
A system consists of an **entry condition** and a set of **output features**.

**Figure:** A fragment of a system network diagram
More than one system may share the same entry condition.

Figure: Simultaneous systems
<table>
<thead>
<tr>
<th>Syntax</th>
<th>Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a \rightarrow [x \ y]$</td>
<td><strong>system</strong> if $a$, then $x$ or $y$ (abbreviated as $a : x</td>
</tr>
<tr>
<td>$a \ b \rightarrow [x \ y]$</td>
<td><strong>disjunction in entry condition</strong> if $a</td>
</tr>
</tbody>
</table>
Syntax and Semantics of a System Network (2/4)

\[
\begin{align*}
\{a, b\} & \quad \rightarrow \quad \begin{bmatrix} x \\ y \end{bmatrix} \\
\{a\} & \quad \rightarrow \quad \begin{bmatrix} x \\ y \\ m \\ n \end{bmatrix}
\end{align*}
\]

- Conjunction in entry condition:
  If \( a \) and \( b \) (abbreviated as \( a \& b \)), then \( x | y \)

- Simultaneity:
  If \( a \), then simultaneously \( x | y \) and \( m | n \)

Paolo Dongilli... that needed Natural Language Support.
### Syntax and Semantics of a System Network (3/4)

<table>
<thead>
<tr>
<th>Rule</th>
<th>Expression</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a \rightarrow { x \rightarrow m, n }$</td>
<td>▶️</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>delicacy ordering</td>
</tr>
<tr>
<td></td>
<td></td>
<td>if $a$, then $x</td>
</tr>
<tr>
<td>$a \leftarrow m \rightarrow \ast$</td>
<td>▶️</td>
<td></td>
</tr>
<tr>
<td>$a \leftarrow n \rightarrow \ast$</td>
<td>▶️</td>
<td></td>
</tr>
<tr>
<td>$a \leftarrow x \ast \rightarrow y$</td>
<td>▶️</td>
<td></td>
</tr>
<tr>
<td>$a \leftarrow m \rightarrow \ast n$</td>
<td>▶️</td>
<td></td>
</tr>
</tbody>
</table>

**Paolo Dongilli**

... that needed Natural Language Support.
gate
(one choice only) if $x$ and $f$, then $m$

recursive system (logical)
if $a$, then $x|y$ and simultaneous option of entering and selecting from the same system again;
$//$ = stop
Realization rules show how the paradigmatic choices in the systems are expressed as syntagmatic chains in the structures of the language.

Figure: System network diagram with realization rules
<table>
<thead>
<tr>
<th>Name</th>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>insert</td>
<td>+Subject</td>
<td>Function inserted as constituent of the structure of the unit being specified</td>
</tr>
<tr>
<td>order</td>
<td>Subject ∧ Finite</td>
<td>One function ordered to precede another</td>
</tr>
<tr>
<td>expand</td>
<td>Mood (Finite)</td>
<td>One function expanded to have another function as constituent</td>
</tr>
<tr>
<td>conflate</td>
<td>Subject / Agent</td>
<td>One function conflated with another one to form the same constituent together</td>
</tr>
<tr>
<td>preselect</td>
<td>Subject: singular</td>
<td>A function preselected for a feature; the realization of the function is constrained to display that feature</td>
</tr>
</tbody>
</table>
Halliday analyzed lexicogrammar into three broad metafunctions:

- **ideational metafunction** is about the natural world in the broadest sense, including our own consciousness, and is concerned with clauses as representations.
- **interpersonal metafunction** is about the social world, especially the relationship between speaker and hearer, and is concerned with clauses as exchanges.
- **textual metafunction** is about the verbal world, especially the flow of information in a text, and is concerned with clauses as messages.
Metafunctions

Halliday analyzed lexicogrammar into three broad metafunctions:

- **ideational metafunction** is about the natural world in the broadest sense, including our own consciousness, and is concerned with clauses as representations.

- **interpersonal metafunction** is about the social world, especially the relationship between speaker and hearer, and is concerned with clauses as exchanges.

- **textual metafunction** is about the verbal world, especially the flow of information in a text, and is concerned with clauses as messages.
Halliday analyzed lexicogrammar into three broad metafunctions:

- **ideational metafunction** is about the natural world in the broadest sense, including our own consciousness, and is concerned with clauses as representations.

- **interpersonal metafunction** is about the social world, especially the relationship between speaker and hearer, and is concerned with clauses as exchanges.

- **textual metafunction** is about the verbal world, especially the flow of information in a text, and is concerned with clauses as messages.
Halliday analyzed lexicogrammar into three broad metafunctions:

- **Ideational metafunction** is about the **natural world** in the broadest sense, including our own consciousness, and is concerned with clauses as representations.

- **Interpersonal metafunction** is about the **social world**, especially the relationship between speaker and hearer, and is concerned with clauses as exchanges.

- **Textual metafunction** is about the **verbal world**, especially the flow of information in a text, and is concerned with clauses as messages.

Paolo Dongilli... that needed Natural Language Support.
Example taken from [Matthiessen and Bateman, 1991]:

<table>
<thead>
<tr>
<th>Theme</th>
<th>Anne</th>
<th>we</th>
<th>’re working</th>
<th>with silver</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vocative</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Locative</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure:** Metafunctional layering
Principal Systems

clauses: ideational

verbal group: interpersonal

nominal group: textual

TRANSITIVITY
TENSE
MODIFICATION
MOOD
MODALITY
PERSON
THEME
VOICE
DETERMINATION

Figure: Principal Systems

[Matthiessen and Bateman, 1991]

Paolo Dongilli . . . that needed Natural Language Support.
Independence of Metafunctions

[Matthiessen and Bateman, 1991]
The Nigel Grammar of English (1/2)

- implementation of the SFG of English
- contains over 600 systems
- starts with the RANK system which selects from the rank scale:
  - clause
  - group/phrase
  - word
  - morpheme
• clause complexes are also handled by this SFG (paratactic and hypotactic relations between two clauses)

• higher-level and more complex structures are handled outside Nigel by systems implementing RST (Rhetorical Structure Theory) [Mann and Thompson, 1988]
[Taboada and Mann, 2006 (1)]
[Taboada and Mann, 2006 (2)]
example of a fully-developed SFG-based generation system

developed at the University of Southern California Information Sciences Institute (USC/ISI) (1980’s)

by Bill Mann, with major contributions by Christian Matthiessen, Sandra Thompson, John Bateman, Robert Kasper, Eduard Hovy and others.

personal participation of Halliday in the development of the Nigel systemic grammar
KPML System

- KPML (KOMET-Penman multilingual)
- development environment for developing and maintaining large-scale sets of multilingual SFL grammars
- multilingual text generation system
- successor of the Nigel/Penman system
The Chooser/Inquiry Interface

Figure: System network diagram with choosers and inquiries
Inquiry Semantics

- developed by William C. Mann (see [Mann, 1983])
- a choice must be made in each disjunctive choice system during grammar network traversal
- each choice system has an associated procedure called its **chooser**, which traverses a decision tree from its root to a single leaf node.
- each branching node has an associated **inquiry**, which obtains information from the external environment in which the grammar is embedded
- the chooser then selects which branch to take according to the response to the inquiry
The Chooser/Inquiry Interface

(system
  :name MOOD-TYPE
  :inputs CLAUSE
  :outputs ((INDICATIVE (INSERT SUBJECT) (INSERT FINITE))
              (IMPERATIVE))
  :chooser MOOD-TYPE-CHOOSER
  :region MOOD
  :metafunction INTERPERSONAL)

(system
  :name INDICATIVE-TYPE
  :inputs INDICATIVE
  :outputs ((DECLARATIVE (ORDER SUBJECT FINITE))
             (INTERROGATIVE))
  :chooser INDICATIVE-TYPE-CHOOSER
  :region MOOD
  :metafunction INTERPERSONAL)

(system
  :name INTERROGATIVE-TYPE
  :inputs INTERROGATIVE
  :outputs ((YES/NO (ORDER FINITE SUBJECT))
            (WH (INSERT WH) (ORDER WH FINITE)))
  :chooser INTERROGATIVE-TYPE-CHOOSER
  :region MOOD
  :metafunction INTERPERSONAL)

Paolo Dongilli... that needed Natural Language Support.
two more ingredients missing

1. the Upper Model
2. an interface between application and generator

Paolo Dongilli

... that needed Natural Language Support.
Introduction
Systemic-Functional Grammar
An SFG-based Generation System

... two more ingredients missing

1. the Upper Model
2. an interface between application and generator
... two more ingredients missing

1. the Upper Model
2. an interface between application and generator
The Upper Model

- a particular domain model based on an ideational grammatical semantic typology for English
- a linguistically motivated ontology
- it reflects English lexical semantics (in fact, English lexicogrammatical semantics)
- it reflects the ideational metafunction, and it is called ideation base
SPL (Sentence Plan Language) represents an interface between the application and the generator [Kasper, 1989]

SPL input includes not only ideational content but also interpersonal and textual specifications.

(p1 / class-ascription
 :domain (A2 / adder
 :identifiability-q identifiable)
 :range (B1 / binary-operator
 :identifiability-q notidentifiable))

→ *The adder is a binary operator.*

...that needed Natural Language Support.
SPL (Sentence Plan Language) represents an interface between the application and the generator [Kasper, 1989]. SPL input includes not only ideational content but also interpersonal and textual specifications.

( pl / class-ascription
  :domain (A2 / adder
    :identifiability-q identifiable)
  :range (B1 / binary-operator
    :identifiability-q notidentifiable))

→ The adder is a binary operator.
SPL (Sentence Plan Language) represents an interface between the application and the generator [Kasper, 1989].

SPL input includes not only ideational content but also interpersonal and textual specifications.

\[
\text{(pl / class-ascription}
\begin{align*}
\text{domain (A2 / adder} & \text{identifiability-q identifiable)} \\
\text{range (B1 / binary-operator} & \text{identifiability-q notidentifiable)} \\
\rightarrow & \text{The adder is a binary operator.}
\end{align*}
\]
Penman-style Architecture

- **Introduction**
- Systemic-Functional Grammar
- An SFG-based Generation System

Paolo Dongilli...that needed Natural Language Support.
I look for an off-roader that runs on diesel. It is sold by a car dealer which is located in Germany, the make is Land Rover and the model is Defender. The off-roader is equipped with A/C, central locking, and leather seats.

I need the price, and the mileage of the off-roader, the name, the city, and the phone number of the car dealer.
Inside the New Query System

Conjunctive Query to SPL Converter

SPL API

Query Interface (QI)

Conjunctive Query
Sentence Plan

Query Expander

NL + concept/phrase mapping
Sentence Plan

DIG API

DIG Compliant Reasoner (FaCT, Pellet, Racer, …)

GUM

Domain Ontology

Domain Lexicon

Basic Lexicon

Natural Language Generator

Nigel Systemic-Functional Grammar of English

Summary

The New User Interface

System Internals

Paolo Dongilli

that needed Natural Language Support.
The car must run on diesel, and the make must be Land Rover.
Critical Task: CQ Partitioning

- Diesel
- Car dealer
- Located in Germany

- Name
- City
- Phone number

- Make: Land Rover
- Model: Defender
- A/C
- Central locking
- Leather seats
- Price
- Mileage

That needed Natural Language Support.
Summary

- Evolution steps of an intelligent query tool
- Information access through ontology navigation
- Introduction to NLG based on computational SFL
- Representation of conjunctive queries in natural language via a SFG-based generator

Outlook
- Mapping algorithm: conjunctive query (CQ) to sentence plan (SPL)
- Refactoring of the generation engine (KPML): J2EE technologies, DIG standard compliance
Summary

- Evolution steps of an intelligent query tool
- Information access through ontology navigation
- Introduction to NLG based on computational SFL
- Representation of conjunctive queries in natural language via a SFG-based generator

Outlook

- Mapping algorithm: conjunctive query (CQ) to sentence plan (SPL)
- Refactoring of the generation engine (KPML): J2EE technologies, DIG standard compliance
Summary

- Evolution steps of an intelligent query tool
- Information access through ontology navigation
- Introduction to NLG based on computational SFL
- Representation of conjunctive queries in natural language via a SFG-based generator

Outlook

- Mapping algorithm: conjunctive query (CQ) to sentence plan (SPL)
- Refactoring of the generation engine (KPML): J2EE technologies, DIG standard compliance
Summary

- Evolution steps of an intelligent query tool
- Information access through ontology navigation
- Introduction to NLG based on computational SFL
- Representation of conjunctive queries in natural language via a SFG-based generator

Outlook

- Mapping algorithm: conjunctive query (CQ) to sentence plan (SPL)
- Refactoring of the generation engine (KPML): J2EE technologies, DIG standard compliance
Summary

- Evolution steps of an intelligent query tool
- Information access through ontology navigation
- Introduction to NLG based on computational SFL
- Representation of conjunctive queries in natural language via a SFG-based generator

Outlook
- Mapping algorithm: conjunctive query (CQ) to sentence plan (SPL)
- Refactoring of the generation engine (KPML): J2EE technologies, DIG standard compliance
Summary

- Evolution steps of an intelligent query tool
- Information access through ontology navigation
- Introduction to NLG based on computational SFL
- Representation of conjunctive queries in natural language via a SFG-based generator

Outlook

- Mapping algorithm: conjunctive query (CQ) to sentence plan (SPL)
- Refactoring of the generation engine (KPML): J2EE technologies, DIG standard compliance
For Further Reading

Paul Piwek, Roger Evans, Lynne Cahill, and Neil Tipper.
Natural Language Generation in the MILE System.

William C. Mann and Sandra A. Thompson.
Rhetorical Structure Theory: Toward a Functional Theory of Text Organization.

Eduard Hovy.
Language generation: overview
*Survey of the state of the art in human language technology*, Chapter 4.1, pages 139–146.

Maite Taboada and William C. Mann
Rhetorical Structure Theory: Looking Back and Moving Ahead.
[Pre-publication version: http://www.sfu.ca/rst/pdfs/Taboada_Mann_RST_Part1.pdf]

Maite Taboada and William C. Mann
Applications of Rhetorical Structure Theory.
[Pre-publication version: http://www.sfu.ca/rst/pdfs/Taboada_Mann_RST_Part2.pdf]
For Further Reading

C. M. I. Matthiessen and J. A. Bateman, editors.  
*Text Generation and Systemic-Functional Linguistics: Experiences from English and Japanese.*  

William C. Mann.  

R. T. Kasper.  
A flexible interface for linking applications to PENMAN's sentence generator.  
In *Proceedings of the DARPA Workshop on Speech and Natural Language*, 1989.

Paolo Dongilli . . . that needed Natural Language Support.