Building Applied Natural Language Generation Systems

Robert Dale and Ehud Reiter

1

What This Tutorial is About

- The design and construction of systems which
  - produce understandable texts in English or other human languages
  - from some underlying non-linguistic representation of information.
- These systems
  - combine knowledge about language and the application domain
  - to automatically produce documents, reports, explanations, help messages, ...

2

Goals of the Tutorial

- For managers
  - to provide a broad overview of the field and what is possible today
- For implementers
  - to provide a realistic assessment of available techniques
- For researchers
  - to highlight the issues that are important in current applied NLG projects

3

Overview

1	An Introduction to NLG
2	Requirements Analysis for NLG
3	NLG Architecture and System Design
4	A Case Study
5	A Closer Look at the Component Tasks
6	Conclusions and Pointers

4

An Introduction to NLG

- What is Natural Language Generation?
- Some Example Systems
- Types of NLG Applications
- When is NLG an Appropriate Technology?

5

What is NLG?

Natural language generation is the process of deliberately constructing a natural language text in order to meet specified communicative goals.

[McDonald 1992]

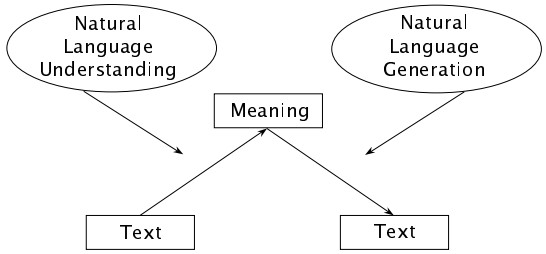
6

What is NLG?

- Goal:
  - computer software which produces understandable texts in English or other human languages
- Input:
  - some underlying non-linguistic representation of information
- Output:
  - documents, reports, explanations, help messages, and other kinds of texts
- Knowledge sources required:
  - knowledge of language and of the domain

7

NLP = NLU + NLG



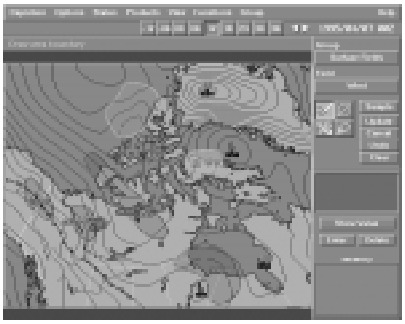
8

Example 1: FoG

- Function:
  - Produces textual weather reports in English and French
- Input:
  - Graphical weather depiction
- User:
  - Environment Canada (Canadian Weather Service)
- Developer:
  - CoGenTex
- Status:
  - Fielded, in operational use since 1992

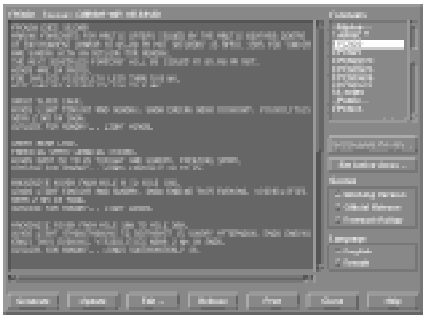
9

FoG: Input



10

FoG: Output



11

Example 2: PlanDoc

- Function:
  - Produces a report describing the simulation options that an engineer has explored
- Input:
  - the simulation log file
- User:
  - Southwest Bell
- Developer:
  - Bellcore and Columbia University
- Status:
  - Fielded, in operational use since 1996

12

PlanDoc: Input

```
RUNID fiberall FIBER 6/19/93 act yes
FA 1301 2 1995
FA 1201 2 1995
FA 1401 2 1995
FA 1501 2 1995
ANF co 1103 2 1995 48
ANF 1201 1301 2 1995 24
ANF 1401 1501 2 1995 24
END. 856.0 670.2
```

13

PlanDoc: Output

This saved fiber refinement includes all DLC changes in RUN-ID ALLDLC. RUN-ID FIBERALL demanded that PLAN activate fiber for CSAs 1201, 1301, 1401 and 1501 in 1995 Q2. It requested the placement of a 48-fiber cable from the CO to section 1103 and the placement of 24-fiber cables from section 1201 to section 1301 and from section 1401 to section 1501 in the second quarter of 1995. For this refinement, the resulting 20 year route PWE was $856.00K, a $64.11K savings over the BASE plan and the resulting 5 year IFC was $670.20K, a $60.55K savings over the BASE plan.

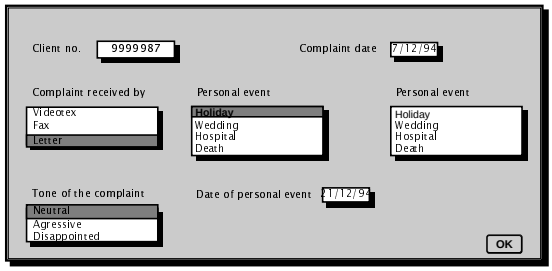
14

Example 3: AlethGen

- Function:
  - Produce a letter (in French) to a customer from a customer-service representative
- Input:
  - Customer database, plus information entered by the service rep with a GUI
- User: La Redoute (French mail-order company)
- Developer: ERLI
- Status:
  - Passed acceptance test, waiting for customer to finish an upgrade of their IT system

15

AlethGen: Input



16

AlethGen: Output

Roubaix, le 12 Décembre 1994

Chère Madame,

Je suis désolée que vous ayez rencontré différents problèmes.

Les taies vertes et l'enveloppe d'édredon imprimée que vous attendez ont été expédiées le 4 Novembre 1994, et notre service de livraison aurait déjà dû les livrer à votre relais de DUNKERQUE. **Je vais me renseigner pour savoir ce qui a pu se produire.** Malheureusement, je ne peux pas renouveler ce linge de lit, car nous ne pourrions pas vous le livrer avant votre départ. Dans votre intérêt, je préfère l'annuler.

Dès aujourd'hui, je régularise votre compte Paiement Confiance de 350,00 F.

D'autre part, vous n'avez pas reçu les chaussures noires. Malheureusement, nous attendons toujours leur rentrée en stock.

Je ne peux que vous demander à nouveau de patienter ou vous conseiller de reporter votre choix sur un autre article de notre catalogue.

Je vous prie de bien vouloir accepter toutes mes excuses.
Je vous souhaite de bonnes vacances.

Bien cordialement.

Nadia, rédactrice

Why Use NLG?

- Important information is stored on computers in ways which are not comprehensible to the end user:
  - graphical weather maps
  - simulation log files
  - databases and spreadsheets
  - expert-system knowledge bases
- NLG systems can present this information to users in an accessible way

18

Two Ways to Use NLG

- Computer as author: the system produces a finished text or document, without human assistance
- Computer as authoring-aid: the system helps a human author produce a text

19

Computer as Author

NLG techniques can be used to:

- generate textual weather forecasts from representations of graphical weather maps
- summarize statistical data extracted from a database or spreadsheet
- explain medical info in a patient-friendly way
- describe a chain of reasoning carried out by an expert system
- paraphrase information in a diagram for inexperienced users

20

Computer as Authoring Aid

NLG systems can be used to:

- help customer service representatives write letters for customers
- help engineers produce management summaries of design paths they have explored
- help personnel officers produce job descriptions
- help technical authors produce instructions for using software
- help an inventor compose a patent claim

21

When are NLG Techniques Appropriate?

Options to consider:

- Text vs Graphics
- Natural Language Generation vs Mail-merge
- Natural Language Generation vs Human Authoring

22

When are NLG Techniques Appropriate?

Factors to consider:

- Is the necessary source data available in a computationally tractable form?
- Is text the right medium?
- How much variation occurs in output texts?
- Is automation justified on the basis of volume, speed requirements or consistency requirements?

23

Overview

24

Requirements Analysis

The developer needs to:

- understand the client's needs
- propose a functionality which addresses these needs

25

Corpus-Based Requirements Analysis

A corpus

- consists of examples of output texts and corresponding input data
- specifies 'by example' the functionality of the proposed NLG system
- is a very useful resource for design as well as requirements analysis

26

Corpus-Based Requirements Analysis

Four Steps:

- assemble an initial corpus of (human-authored) output texts and associated input data
- analyze the information content of the corpus texts in terms of the input data
- develop a target text corpus
- create a formal functional specification

27

Step 1: Creating an Initial Corpus

- Collect a corpus of input data and associated (human-authored) output texts
- One source is archived examples of human-authored texts
- If no human-authored examples of the required texts exist, ask domain experts to produce examples
- The corpus should cover the full range of texts expected to be produced by the NLG system

28

Problems with the Initial Corpus

A quick scan of the corpus may reveal that

- it is impossible to automatically generate some of the texts in the initial corpus
- the corpus texts appear sub-optimal and open to improvement
- texts written by different human authors have very different structures and styles
- it is not obvious why a particular text was generated for a particular input

29

Step 2: Analyzing the Content of the Texts

- Goal:
  - to determine where the information present in the texts comes from, and the extent to which the proposed NLG system will have to manipulate this information
- Result:
  - a detailed understanding of the correspondences between the available input data and the output texts in the initial corpus

30

An Example

- Context: we want to build an NLG system which replies to rail-travel inquiries.
- The system has access to train timetable information

31

An Example Text

There are 20 trains daily from Aberdeen to Glasgow. The next train is the Caledonian Express. It leaves Aberdeen at 10am. This train has been running for 55 years.
Thank you for considering rail travel.

32

Units of Information

- The first step in analyzing the corpus for information content is to identify the individual units of information that make up each of the texts
- A general rule of thumb: units of information correspond to individual clauses or sentences

33

Units of Information

1 There are 20 trains daily from Aberdeen to Glasgow.
2 The next train is the Caledonian Express.
3 It leaves Aberdeen at 10am.
4 This train has been running for 55 years.
5 Thank you for considering rail travel.

34

Information Types in Text

- Unchanging text
- Directly-available data
- Computable data
- Unavailable data

35

Unchanging Text

There are 20 trains daily from Aberdeen to Glasgow. The next train is the Caledonian Express. It leaves Aberdeen at 10am. This train has been running for 55 years.
Thank you for considering rail travel.

36

Directly-Available Data

There are 20 trains daily from Aberdeen to Glasgow. The next train is the Caledonian Express. It leaves Aberdeen at 10am. This train has been running for 55 years.

Thank you for considering rail travel.

37

Computable Data

There are 20 trains daily from Aberdeen to Glasgow. The next train is the Caledonian Express. It leaves Aberdeen at 10am. This train has been running for 55 years.

Thank you for considering rail travel.

38

Unavailable Data

There are 20 trains daily from Aberdeen to Glasgow. The next train is the Caledonian Express. It leaves Aberdeen at 10am. This train has been running for 55 years.

Thank you for considering rail travel.

39

Solving the Problem of Unavailable Data

- More information can be made available to the system: this may be expensive
- If the system is an authoring-aid, a human author can add this information
- The corpus can be revised to eliminate clauses that convey this information

40

Step 3: Building the Target Text Corpus

Mandatory changes:

- eliminate unavailable data
- specify what text portions will be human-authored

Optional changes:

- simplify the text to make it easier to generate
- improve human-authored texts
- enforce consistency between human authors

41

Step 4: Functional Specification

- Based on an agreed target text corpus
- Explicitly states role of human authoring, if present at all
- Explicitly states structure and range of inputs to be used

42

Research Issues

- Development of an appropriate corpus-analysis methodology
- Using expert-system knowledge-acquisition techniques
- Automating aspects of corpus analysis
- Integrating corpus-analysis with standard requirements analysis procedures

43

Overview

1 An Introduction to NLG
2 Requirements Analysis for NLG
3 NLG Architecture and System Design
4 A Case Study
5 A Closer Look at the Component Tasks
6 Conclusions and Pointers

44

NLG System Architecture

- The Component Tasks in NLG
- A Standard Architecture
- Alternatives Architectures

45

Component Tasks in NLG

1 Content determination
2 Discourse planning
3 Sentence aggregation
4 Lexicalisation
5 Referring expression generation
6 Syntactic and morphological realization
7 Orthographic realization

46

1 Content Determination

- The process of deciding what to say
- Can be viewed as the construction of a set of MESSAGES from the underlying data source
- Messages are aggregations of data that are appropriate for linguistic expression: each may correspond to the meaning of a word or a phrase
- Messages are based on domain entities, concepts, and relations

47

Some Messages

IDENTITY(NEXTTRAIN, CALEDONIANEXPRESS)
; The next train is the Caledonian Express
DEPARTURETIME(CALEDONIANEXPRESS, 1000).
;The Caledonian Express leaves at 10am
COUNT((TRAIN, SOURCE(ABERDEEN),
DESTINATION(GLASGOW)), 20, PERDAY)
;There are 20 trains daily from Aberdeen to Glasgow

48

2 Discourse Planning

- A text is not just a random collection of sentences
- Texts have an underlying structure in which the parts are related together
- Two related issues:
  - conceptual grouping
  - rhetorical relationships

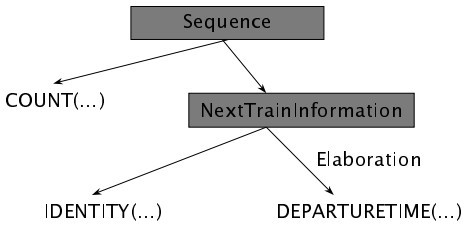
49

The Example Text

There are 20 trains daily from Aberdeen to Glasgow. The next train is the Caledonian Express. It leaves Aberdeen at 10am.

50

A Text Plan



51

3 Sentence Aggregation

- A one-to-one mapping from messages to sentences results in disfluent text
- Messages need to be combined to produce larger and more complex sentences
- The result is a sentence specification or SENTENCE PLAN

52

An Example of Sentence Aggregation

- Without aggregation:
  - The next train is the Caledonian Express.
    It leaves Aberdeen at 10am.
- With aggregation:
  - The next train, which leaves at 10am, is the Caledonian Express.

53

4 Lexicalisation

- So far we have determined text content and the structuring of the information into paragraphs and sentences
- Lexicalisation determines the particular words to be used to express domain concepts and relations

54

Lexicalisation

- In our example, should the DEPARTURETIME relation be expressed using the verb leave or depart?
- How do we express different nuances of meaning?
- What words should be used in different languages?

55

5 Referring Expression Generation

- Referring expression generation is concerned with how we describe domain entities in such a way that the hearer will know what we are talking about
- Major issue is avoiding ambiguity
- Fluency pulls in the opposite direction

56

Kinds of Referring Expressions

- Proper names
  - Aberdeen, Scotland
  - Aberdeen
- Definite Descriptions
  - the train that leaves at 10am
  - the next train
- Pronouns
  - it

57

6 Syntactic and Morphological Realization

- Every natural language has grammatical rules that govern how words and sentences are constructed
- Morphology: rules of word formation
- Syntax: rules of sentence formation

58

Morphological Realization

Rules like:

- to form the past tense of a verb add ed
  - walk + ed = walked
- to form the plural of a noun add s
  - train + s = trains
- if a root ends in e and the suffix starts with a vowel, delete the final e in the root
  - like + ed = liked

59

Syntactic Realization

Rules like:

- the subject goes before the verb
- the subject and verb should agree in number

60

7 Orthographic Realization

- Orthographic realization is concerned with matters like casing and punctuation
- This also extends into typographic issues: font size, column width ...

61

Orthographic Rules

Rules like:

- sentences begin with upper case letters
- sentences end in full stops
- if the last word in a sentence is an abbreviation that ends in a full stop, then this is merged with the full stop at the end of the sentence

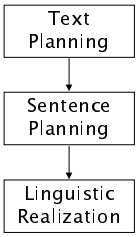
62

Tasks and Architecture in NLG

• Content determination • Discourse planning	Text Planning
• Sentence aggregation • Lexicalisation • Referring expression generation	Sentence Planning
• Syntax + morphology • Orthographic realization	Linguistic Realization

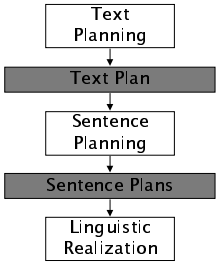
63

A Pipelined Architecture



64

Intermediate Representations



65

A Text Plan

```
((type textplan)
 (relations ((sequence satellite-01 satellite02)))
 (satellite-01 ((nucleus ((message-id msg091)
                          (process exists)
                          (args ((object train)
                              (source aberdeen)
                              (destination glasgow)
                              (frequency ((number 20) (period day)))))))))
 (satellite-02 ((relations ((elaboration nucleus satellite-01)))
                (nucleus ((message-id msg092)
                          (process identity)
                          (args ((arg1 next-train)
                                 (arg2 cal-express)))))
                (satellite-01 ((nucleus ((message-id msg093)
                                         (process departure)
                                         (args ((object cal-express)
                                                (time 1000))))))))))
```

66

A Sentence Plan

```
(S1/ThereBe
   :object (O1/train
            :cardinality 20
            :relations
              ((R1/period :value daily)
               (R2/source :value Aberdeen)
               (R3/destination :value Glasgow))))
```

There are 20 trains daily from Aberdeen to Glasgow.

67

Other Architectures

Variants on the 'standard' architecture:

- shift tasks around
  - for example, include lexicalisation in the realizer
- allow feedback between stages

68

Integrated Architectures

Integrate everything into one reasoner:

- represent all tasks in the same way: eg as constraints, axioms, plan operators ...
- feed specifications into a constraint-solver, theorem-prover ...

Pros and Cons

- For: theoretically very elegant
- For: good support for interdependencies between tasks
- Against: absence of modularisation makes it very expensive in engineering terms

69

Research Questions

- How do the different tasks interact?
- Is there an architecture which combines the theoretical elegance of integrated approaches with the engineering simplicity of the pipeline?
- How should text and sentence plans be represented?
- How should multimodal documents be represented?

70

Overview

1 An Introduction to NLG
2 Requirements Analysis for NLG
3 NLG Architecture and System Design
4 A Case Study
5 A Closer Look at the Component Tasks
6 Conclusions and Pointers

71

A Case Study in Applied NLG

- Each month an institutional newsletter publishes a summary of the month's weather
- The summaries are based on automatically collected meteorological data
- The person who writes these summaries will no longer be able to
- The institution wants to continue publishing the reports and so is interested in using NLG techniques to do so

72

A Weather Summary

MARSFIELD (Macquarie University No 1)
On Campus, Square F9

TEMPERATURES (C)

Mean Max for Mth:	18.1 Warmer than average
Mean Max for June (20 yrs):	17.2
Highest Max (Warmest Day):	23.9 on 01
Lowest Max (Coldest Day):	13. On 12
Mean Min for Mth:	08.2 Much warmer than ave
Mean Min for June (20 yrs):	06.4
Lowest Min (Coldest Night):	02.6 on 09
Highest Min (Warmest Night):	13.5 on 24

RAINFALL (mm) (24 hrs to 09:00)

Total Rain for Mth:	90.4 on 12 days. Slightly below average.
Wettest Day (24h to 09:00):	26.4 on 11
Average for June (25 yrs):	109.0 on 10
Total for 06 mths so far:	542.0 on 72 days. Very depleted.
Average for 06 mths (25 yrs):	762.0 on 71 days
Annual Average Rainfall (25 yrs):	1142.8 on 131 days

WIND RUN (at 2m height) (km) (24 hrs to 09:00)

Total Wind Run for Mth:	1660
Windiest Day (24 hrs to 09:00):	189 on 24, 185 on 26, 172 on 27
Calmest Day (24 hrs to 09:00):	09 on 16

SUNRISE & SUNSET

Date	Sunrise	Sunset	Difference
01 Jun	06:52	16:54	10:02
11 Jun	06:57	16:53	09:56
21 Jun	07:00	16:54	09:54
30 Jun	07:01	16:57	09:56

(Sunset times began to get later after about June 11)
(Sunrise times continue to get later until early July)
(Soon we can take advantage of the later sunsets)

SUMMARY

The month was warmer than average with average rainfall, but the total rain so far for the year is still very depleted. The month began with mild to warm maximums, and became cooler as the month progressed, with some very cold nights such as June 09 with 02.6. Some other years have had much colder June nights than this, and minimums below zero in June are not very unusual. The month was mostly calm, but strong winds blew on 23, 24 and 26, 27. Fog occurred on 17, 18 after some rain on 17, heavy rain fell on 11 June.

73

A Weather Summary

The month was warmer than average with average rainfall, but the total rain so far for the year is still very depleted. The month began with mild to warm maximums, and became cooler as the month progressed, with some very cold nights such as June 09 with 02.6. Some other years have had much colder June nights than this, and minimums below zero in June are not very unusual. The month was mostly calm, but strong winds blew on 23, 24 and 26, 27. Fog occurred on 17, 18 after some rain on 17, heavy rain fell on 11 June.

74

The Input Data

- A set of 16 data elements collected automatically every 15 minutes: air pressure, temperature, wind speed, rainfall ...
- Preprocessed to construct DailyWeatherRecords:

```
((type dailyweatherrecord)
    (date ((day ...)
           (month ...)
           (year ...)))
    (temperature ((minimum ((unit degrees-centigrade)
                            (number ...)))
                  (maximum ((unit degrees-centrigrade)
                            (number ...)))))
    (rainfall ((unit millimetres)
               (number ...))))
```

75

Other Available Data

- Historical Data: Average temperature and rainfall figures for each month in the Period of Record (1971 to present)
- Historical Averages: Average values for temperature and rainfall for the twelve months of the year over the period of record

76

Building the Target Texts

For each text:

- Omit information that is not available in a computationally tractable form
- Assume we don't have wind, sunshine, thunderstorm and mist or fog information
- Then, edit the results for fluency

77

Initial Text #1

The month was cooler and drier than average, with the average number of rain days. The total rain for the year so far is well below average. There was rain on every day for 8 days from 11th to 18th, with mist and fog patches on 16th and 17th. Rainfall amounts were mostly small, with light winds.

78

Text #1: Simplifications

The month was cooler and drier than average, with the average number of rain days. The total rain for the year so far is well below average. There was rain on every day for 8 days from 11th to 18th, with mist and fog patches on 16th and 17th. Rainfall amounts were mostly small, with light winds.

79

Text #1: Simplified Text

The month was cooler and drier than average, with the average number of rain days. The total rain for the year so far is well below average. There was rain on every day for 8 days from 11th to 18th. Rainfall amounts were mostly small.

80

Target Text #1

The month was cooler and drier than average, with the average number of rain days, but the total rain for the year so far is well below average. Although there was rain on every day for 8 days from 11th to 18th, rainfall amounts were mostly small.

81

Initial Text #2

The month was rather dry with only three days of rain in the middle of the month. The total for the year so far is very depleted again, after almost catching up during March. Mars Creek dried up again on 30th April at the waterfall, but resumed on 1st May 1 light rain. This is the fourth time it dried up this year.

82

Text #2: Simplifications

The month was rather dry with only three days of rain in the middle of the month. The total for the year so far is very depleted again, after almost catching up during March. Mars Creek dried up again on 30th April at the waterfall, but resumed on 1st May 1 light rain. This is the fourth time it dried up this year.

83

Target Text #2

The month was rather dry with only three days of rain in the middle of the month. The total for the year so far is very depleted again.

84

Initial Text #3

The month was our driest and warmest August in our 24 year record, and our first 'rainless' month. The 26th August was our warmest August day in our record with 30.1, and our first 'hot' August day (30). The month forms part of our longest dry spell 47 days from 18 July to 02 September 1995. Rainfall so far is the same as at the end of July but now is very deficient.

85

Target Text #3

The month was our driest and warmest August in our 24 year record, and our first rainless month. The 26th August was our warmest August day in our record with 30.1, and our first hot August day. Rainfall is now very deficient.

86

The Case Study So Far

We'll assume that:

- We have located the source data
- We have preprocessed the data to build the DailyWeatherRecords
- We have constructed an initial corpus of texts
- We have modified the initial corpus to produce a set of target texts

87

Is it Worth Using NLG?

- For one summary a month probably not, especially given the simplifications required to the texts to make them easy to generate
- However, the client is interested in a pilot study:
  - in the future there may be a shift to weekly summaries
  - there are many automatic weather data collection sites each of which could use the technology

88

Overview

1 An Introduction to NLG
2 Requirements Analysis for NLG
3 NLG Architecture and System Design
4 A Case Study
5 A Closer Look at the Component Tasks
6 Conclusions and Pointers

89

Component Tasks in NLG

- Text Planning
  - Content determination
  - Discourse planning
- Sentence Planning
  - Sentence aggregation
  - Lexicalisation
  - Referring expression generation
- Linguistic Realization
  - Syntactic and morphological realization
  - Orthographic realization

90

Text Planning

Goal:

- To determine what messages to communicate, and how to rhetorically structure these messages

Approaches

- Methods based on reasoning about intention
- Methods based on observations about text structure
- More general expert system-based methods

91

Methods based on Reasoning about Intentions

Typically use AI planning techniques:

- Goal = desired communicative effect
- Plan constituents = things the system can say (speech acts)
- involves reasoning about user's beliefs to determine what to say and how to rhetorically structure that information
- requires reasoning about why information should be included

92

Methods based on Reasoning about Intentions

Analysis:

- principled and theoretically elegant
- expensive and complex
- may be most appropriate for dialog systems
- not widely used in applied NLG systems

93

Methods based on Text Structure

Basic idea:

- texts often follow conventionalised patterns
- these patterns can be captured by means of 'text grammars' that dictate both content and coherent structure
- these effectively recompile some of the reasoning that has to be carried out explicitly in more sophisticated methods

94

Methods based on Text Structure

Text formats encapsulated in SCHEMAS:

- template programs which produce text plans
- specify how a particular text plan can be constructed using smaller schemas or atomic messages
- can specify many degrees of variability and optionality
- may include content determination or only text structuring

95

Methods based on Text Structure

Implementing schemas:

- simple schemas can be expressed as grammars
- more flexible schemas usually implemented as macros or class libraries on top of a conventional programming language, where each schema is a procedure
- usually embody ideas from Rhetorical Structure Theory to provide text coherence
- currently most popular text-planning approach in applied NLG

96

An Example Schema

```
Schema-name: PreviousAttemptFailedParagraph()
condition: PreviousAttemptToQuit == TRUE
body:
  N1 = SummarisePreviousAttempts();
  N2 = LookForEncouragingSigns();
  N3 = Message("Most people who successfully quit
  smoking make several unsuccessful attempts
  first …")
  If (N2 == NULL)
      return Contrast(N1, N3)
  else return Contrast(N1, Elaboration(N2, N3))
```

97

Deriving Schemas from a Corpus

Using the Target Text Corpus:

- take a small number of similar corpus texts
- break up these texts into messages, and try to determine how each message can be computed from the input data
- propose rules or structures which explain why message *x* is in text A but not text B -- this may be easier if messages are organised into a taxonomy
- discuss this analysis with domain experts, and iterate
- repeat the exercise with a larger set of corpus texts

98

Schemas: Pros and Cons

Advantages of schemas:

- Computationally efficient
- Allow arbitrary computation when necessary
- Naturally support genre conventions
- Relatively easy to acquire from a corpus

Disadvantages

- Less flexible
- May be difficult to reuse in other applications

99

Expert System Approaches

Use (constructive) expert-system techniques
- production rules
- case-based reasoning

Analysis
- Places planning and schemas into perspective
- Suggests many other possible algorithms

100

A Production Rule Example

```
if (and (?deposit < ?total_cost)
            (case :disputed-charges))
then
    (add-node
    (rhetorical fact-explanation
      (message ?service_provider has submitted an
        additional charge to your account)
      (message ?deposit was less than
        ?total_cost)))
```

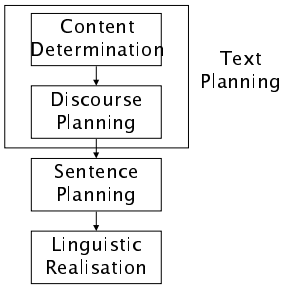
101

Research Issues

- Using other ideas from expert systems -- eg case based reasoning
- Principled ways of integrating shallow and deep reasoning
- A better understanding of rhetorical relations

102

The WeatherReporter Architecture



103

Text Planning

Content Determination:

- Build messages for the reporting period

Discourse Planning

- Organize messages to be presented in the text

104

Content Determination

Determining Message Types:

- Break texts into largest possible base informational elements
- Identify underlying data required for each kind of informational element
- define a set of message types that collect together underlying data in ways that are convenient for linguistic expression

105

Content Determination

The Key Issues in Message Specification:

- determine the most useful clusterings of information
- represent these clusterings in a way that makes manipulation for linguistic expression easy

106

Content Determination

- Routine messages
  - MonthlyRainFallMsg, MonthlyTemperatureMsg, RainSoFarMsg, MonthlyRainyDaysMsg
- Always constructed for any summary to be generated

107

Content Determination

- Significant Event messages
  - RainEventMsg, RainSpellMsg, TemperatureEventMsg, TemperatureSpellMsg
- Only constructed if the data warrants their construction: eg if rain occurs on more than a specified number of days in a row

108

Content Determination

A MonthlyRainfallMsg:

```
((message-id msg091)
  (message-type monthlyrainfall)
  (period ((month 04)
           (year 1996)))
  (absolute-or-relative relative-to-average)
  (relative-difference ((magnitude ((unit millimeters)
                                    (number 4)))
                        (direction +))))
```

109

Content Determination

A RainSpellMsg:

```
((message-id msg096)
 (message-type rainspellmsg)
 (period ((begin ((day 04)
                  (month 02)
                  (year 1995)))
          (end ((day 11)
                (month 02)
                (year 1995)))
          (duration ((unit day)
                     (number 8)))))
 (amount ((unit millimetres)
          (number 120))))
```

110

Discourse Planning

- The content of the text to be generated has been determined by the message building phase
- The goal of discourse planning is to assemble this set of messages into a coherent text

111

Discourse Planning

A Simple Discourse Schema:

WeatherSummary →
- MonthlyTempMsg
- MonthlyRainfallMsg
- RainyDaysMsg
- RainSoFarMsg

112

Discourse Planning

Limitations of a simple schema-based approach:
- Very rigid
- Doesn't allow for special circumstances

Ways of adding more flexibility:
- add optionality to the schema to deal with special cases
- segment knowledge of what makes a text coherent into separate rules

113

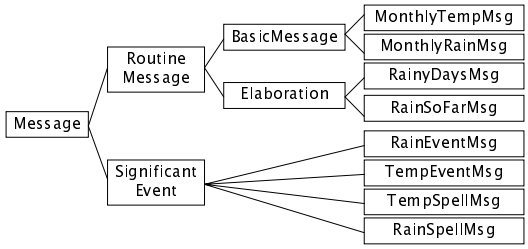
Discourse Planning

Three basic rhetorical relationships:
- SEQUENCE
- ELABORATION
- CONTRAST

Applicability of rhetorically-based planning operators determined by Message Status and Message Topic

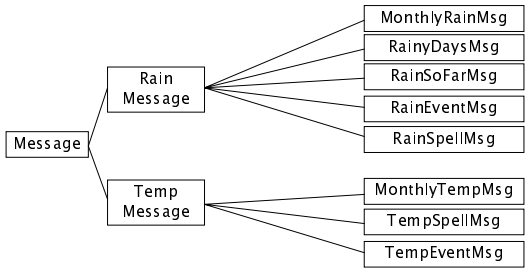
114

Message Status



115

Message Topic



116

Discourse Planning

SEQUENCE

- Two messages can be connected by a SEQUENCE relationship if they are both BasicMessages

117

Discourse Planning

ELABORATION

- Two messages can be connected by an ELABORATION relationship if:
  - they are both have the same TOPIC
  - the nucleus is a BasicMessage

118

Discourse Planning

CONTRAST

- Two messages can be connected by a CONTRAST relationship if:
  - they both have the same TYPE
  - they both have the feature `absolute-or-relative = relative-to-average`
  - they have different values for `relative-difference:direction`

119

Discourse Planning

- Select a start message
- Use rhetorical relation operators to add messages to this structure until all messages are consumed or no more operators apply
- Start message is any routine message

120

Discourse Planning

The algorithm:

TextPlan = StartMessage
MessageSet = MessageSet - StartMessage
repeat
- find a rhetorical operator that will allow attachment of a message to the TextPlan
- attach message and remove from MessageSet

until MessageSet = 0 or no operators apply

121

Target Text #1

The month was cooler and drier than average, with the average number of rain days, but the total rain for the year so far is well below average. Although there was rain on every day for 8 days from 11th to 18th, rainfall amounts were mostly small.

122

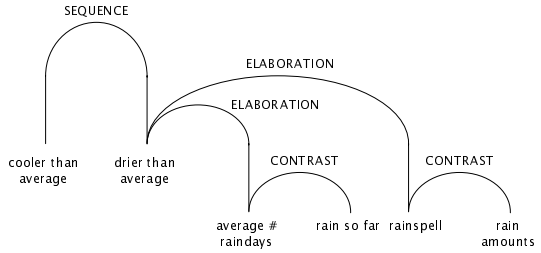
Discourse Planning

The Message Set:

CoolerThanAverage
DrierThanAverage
AverageNumberOfRainDays
RainSoFar
RainSpell
RainAmounts

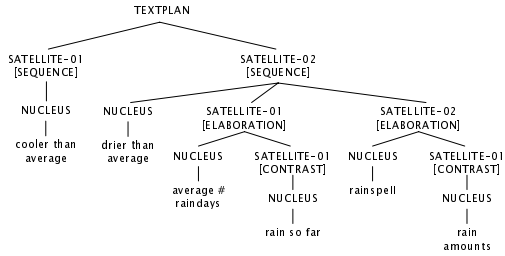
123

Discourse Planning



124

A Text Plan



125

A Text Plan

```
((type textplan)
 (relations ((sequence satellite-01 satellite02)))
 (satellite-01 ((nucleus (MONTHLYTEMPMSG))))
 (satellite-02 ((nucleus (MONTHLYRAINFALLMSG))
                (relations ((elaboration nucleus satellite-01)
                            (elaboration nucleus satellite-02)))
                (satellite-01 ((relations ((contrast nucleus satellite-01))
                               (nucleus (RAINYDAYSMSG))
                               (satellite-01 ((nucleus (RAINSOFARMSG))))))
                (satellite-02 ((relations ((contrast nucleus satellite-01))
                               (nucleus (RAINSPELLMSG))
                               (satellite-01 ((nucleus (RAINAMOUNTSMSG))))))))))
```

126

Discourse Planning

- Result is a TEXT PLAN: a tree structure populated by messages at its leaf nodes
- Next step: realising the messages as text

127

A Simple Realiser

- We can produce one output sentence per message in the discourse plan
- A specialist fragment of code for each message type determines how that message type is realised

128

A Simple Realiser

For the MonthlyTemperatureMsg:

```
TempString = case (TEMP - AVERAGETEMP)
   [2.0 … 2.9]:  ‘ very much warmer than average.’
   [1.0 … 1.9]:  ‘much warmer than average.’
   [0.1 … 0.9]:  ‘slightly warmer than average.’
   [-0.1 … -0.9]:  ‘slightly cooler than average.’
   [-1.0 … -1.9]:  ‘much cooler than average.’
   [-2.0 … -2.9]:  ‘ very much cooler than average.’
endcase
Sentence = ‘The month was’ + TempString
```

129

One Message per Sentence

1 The month was cooler than average.
2 The month was drier than average.
3 There were the average number of rain days.
4 The total rain for the year so far is well below average.
5 There was rain on every day for 8 days from 11th to 18th.
6 Rainfall amounts were mostly small.

130

Target Text #1

The month was cooler and drier than average, with the average number of rain days, but the total rain for the year so far is well below average. Although there was rain on every day for 8 days from 11th to 18th, rainfall amounts were mostly small.

131

Simple Templates

Problems with simple templates:

- MonthlyTemp and MonthlyRainfall don't always appear in the same sentence
- When they do appear in the same sentence, they don't always appear in the same order
- Each can be realised in different ways: eg 'very warm' vs 'warmer than average'
- Additional information may or may not be incorporated into the same sentence

132

Sentence Planning

Goal:
- To convert a text plan into a sequence of sentence definitions

Tasks:
- Aggregation
- Lexicalisation
- Reference

133

Aggregation

Sentence aggregation:
- combining messages into sentences

Paragraph aggregation:
- combining sentences into paragraphs

Paragraph aggregation is poorly understood

134

Ways of Aggregating Messages

Some possibilities:
- Simple conjunction
- Ellipsis
- Embedding
- Set introduction

135

Simple Conjunction

Without aggregation:

- The next train is the Caledonian Express.
- It leaves at 10AM.

With aggregation:

- The next train is the Caledonian Express, and it leaves at 10AM.

136

Ellipsis

Without aggregation:

- It leaves at 10AM.
- It arrives at 12.30PM.

With aggregation:

- It leaves at 10AM, and Ø arrives at 12.30PM.

137

Embedding

Without aggregation:

- The next train is the Caledonian Express.
- It leaves at 10AM.

With aggregation:

- The next train, which leaves at 10AM, is the Caledonian Express.

138

Set Introduction

Without aggregation:

- It has a snack bar.
- It has a restaurant car.

With aggregation:

- It has {a snack bar, a restaurant car}.
- It has a snack bar and a restaurant car.

Caution: need to avoid changing the meaning:

- John bought a TV and Bill bought a TV.
- John and Bill bought a TV.

139

Choosing Aggregations

Usually many ways to aggregate a given message set:

- The next train is the Caledonian Express. It leaves at 10am
- The next train is the Caledonian Express, and it leaves at 10am.
- The next train, which leaves at 10am, is the Caledonian Express.
- The next train is the Caledonian Express, which leaves at 10am.

Which is best?

140

Choice Heuristics

A few suggestions from the literature:

- conform to genre conventions and rules
- only aggregate messages which are siblings in the text plan tree
- only use relative clauses when the messages are linked by an Elaboration rhetorical relation

More research is needed

141

Lexicalisation

Converting concepts into words
- It departs at 10AM
- It leaves at 10AM
- It has a departure time of 10AM

Assume concepts are 'close' to words:
- Concept is DepartureTime, not a record of the train's location and velocity at 0958, 0959, 1000, 1001, 1002
- latter case is very poorly understood

142

Lexicalisation Heuristics

Some suggestions from the literature
- Enforce variation -- don't overuse words
- Discourse focus:
  - John sold Mary an apple
  - Mary bought an apple from John
- Conform to level of formality and other pragmatic criteria:
  - father vs dad
- Conform to genre convention and rules:
  - Paracetamol vs Tylenol vs Acetaminophen
- Express multiple concepts with a single word:
  - He is a terrorist = kills_people & bad_person

143

Lexicalisation Algorithms

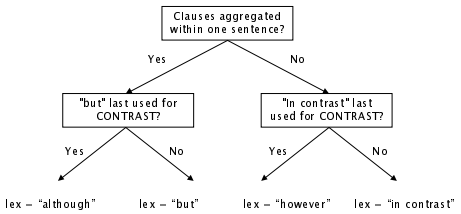
Decision trees:
- choose between ways of expressing a single concept, based on simple questions
- very simple and easy to implement, probably the most popular approach

Graph-rewriting:
- treat each message as a semantic network, and rewrite it according to a concept→word dictionary
- more powerful, theoretically elegant
- perhaps only justified in systems with multilingual output

144

A Discrimination Net

For realizing the CONTRAST rhetorical relation:



145

Reference

How can we identify specific domain objects and entities?

- Initial introduction of an object
- Subsequent references to an already salient object

146

Initial Reference

Introducing an object into the discourse:

- use a full name
  - the Caledonian Express
- relate to an object that is already salient
  - the train's restaurant car
- specify physical location
  - the train behind the Hertz billboard

Poorly understood; more research is needed

147

Subsequent Reference

Some possibilities:

- Pronouns
  - It leaves at 10am
- Definite NPs
  - The train leaves at 10am
- Proper names, possibly abbreviated
  - The Caledonian leaves at 10am

148

Choosing a Form of Reference

Some suggestions from the literature

- use a pronoun if it refers to an entity mentioned in the previous clause, and there is no other entity in the previous clause that the pronoun could refer to
- otherwise use a name, if a short one exists
- otherwise use a definite NP
- also important to conform to genre conventions -- for example, there are more pronouns in newspaper articles than in technical manuals

149

Example

I am taking the Caledonian Express tomorrow. It is a much better train than the Grampian Express. The Caledonian has a real restaurant car, while the Grampian just has a snack bar. The restaurant car serves wonderful fish, while the snack bar serves microwaved mush.

150

Sentence Planning: Choice

Choice may be the central issue in sentence planning:

- which aggregation, lexicalisation, or referring expression should be used?
- clear that genre/sublanguage is important in all of these choices
- evidence and support may come from psycholinguistics and practitioner's wisdom

151

Another Choice Example

Which is better?

1 Press the button if the blue light comes on.
2 If the blue light comes on, press the button.

Considerations

- The psychologist's view: (1) is read quicker
- The technical author's view: (2) is less likely to be misinterpreted

152

Research Issues

- How do we make choices?
- How do we perform higher-level aggregation, such as forming paragraphs from sentences?
- How do we lexicalise if domain concepts do not easily map into words?
- What is the best way of making an initial reference to an object?

153

Sentence Planning

- Examine the target texts to determine the range of expression required
- Construct mappings from the text plan to sentence plans that can be given to a realiser

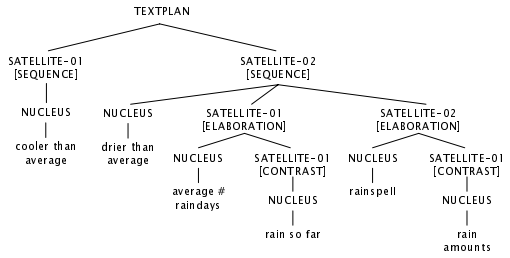
154

Sentence Planning

- Sentence aggregation rules work within substructures of the text plan
- The rules identify structures that can be combined
- Rules may combine entire messages or parts of messages

155

A Text Plan



156

A Text Plan

```
((type textplan)
 (relations ((sequence satellite-01 satellite02)))
 (satellite-01 ((nucleus (MONTHLYTEMPMSG))))
 (satellite-02 ((nucleus (MONTHLYRAINFALLMSG))
                (relations ((elaboration nucleus satellite-01)
                            (elaboration nucleus satellite-02)))
                (satellite-01 ((relations ((contrast nucleus satellite-01)))
                               (nucleus (RAINYDAYSMSG))
                               (satellite-01 ((nucleus (RAINSOFARMSG))))))
                (satellite-02 ((relations ((contrast nucleus satellite-01)))
                               (nucleus (RAINSPELLMSG))
                               (satellite-01 ((nucleus (RAINAMOUNTSMSG))))))))
```

157

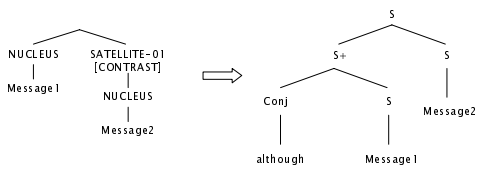
Sentence Planning Rules

Sentence planning rules are sensitive to rhetorical relations:

- If two messages are in a SEQUENCE relation they can be conjoined at the same level
- If one message is an ELABORATION of another it can either be conjoined at the same level or embedded as a minor clause or phrase
- If one message is a CONTRAST to another it can be conjoined at the same level or embedded as a minor clause or phrase

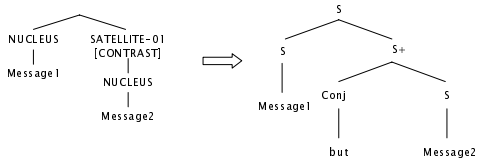
158

A Sentence Planning Rule



159

A Sentence Planning Rule



160

Variations in Describing Rainfall

Variations in syntactic category:

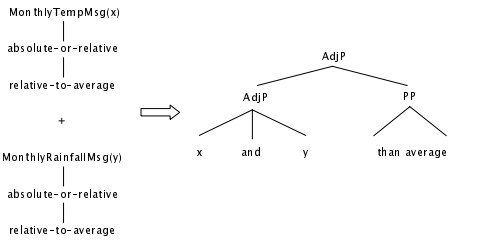
S: [rainfall was very poor indeed]
NP: [a much worse than average rainfall]
AP: [much drier than average]

Variations in semantics:

Absolute: [very dry]
[a very poor rainfall]
Comparative: [a much worse than average rainfall]
[much drier than average]

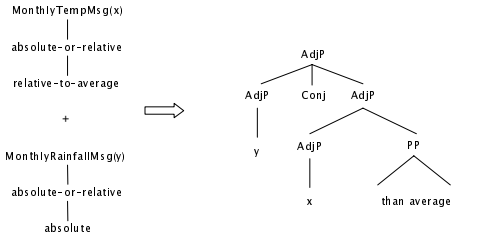
161

A Sentence Planning Rule



162

A Sentence Planning Rule



163

Sentence Planning Rules

Many different results are possible:

- The month was cooler and drier than average. There were the average number of rain days, but the total rain for the year so far is well below average. There was rain on every day for 8 days from 11th to 18th, but rainfall amounts were mostly small.
- The month was cooler and drier than average. Although the total rain for the year so far is well below average, there were the average number of rain days. There was a small mount of rain on every day for 8 days from 11th to 18th.

164

Realisation

Goal:

to convert sentence plans into actual text

Purpose:

to hide the peculiarities of English (or whatever the target language is) from the rest of the NLG system

165

Realisation Tasks

- Insert function words
- Choose correct inflection of content words
- Order words within a sentence
- Apply orthographic rules

166

Realisation

Techniques

- Bi-directional Grammar Specifications
- Grammar Specifications tuned for Generation
- Templates

167

Realisation vs Parsing

Realisation is easier than parsing:

- no need to handle the full range of syntax that a human might use
- no need to resolve ambiguities
- no need to cater for ill-formed input

168

Bi-directional Grammar Specifications

- Key idea: one grammar specification used for both realisation and parsing
- Generally expressed as a declarative set of correspondences between semantic and syntactic structures
- Different processes applied for generation and analysis

169

Bi-directional Grammar Specifications

- Variety of algorithms, including semantic-head driven
- Algorithms often perform lexicalisation as well as realisation
- Theoretically elegant approach
- To date, sometimes used in machine-translation systems, but almost never used in other applied NLG systems

170

Problems with the Bi-directional Approach

- Output of an NLU parser (a semantic form) is very different from the input to an NLG realiser (a sentence plan)
- Debatable whether lexicalisation should be integrated with realisation
- Difficult in practice to engineer large bidirectional grammars
- Difficulties handling fixed phrases

171

Grammar Specifications tuned for Generation

- Grammar provides a set of choices for realisation
- Choices are made on the basis of the input sentence plan
- Grammar can *only* be used for NLG
- Widely used in practice (including FoG, PlanDoc, and AlethGen)
- Working software is available

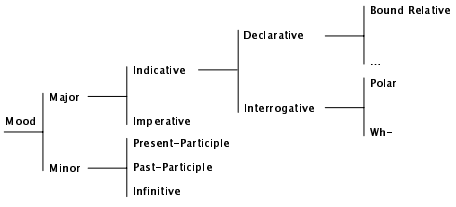
172

Systemic Grammar

- Emphasises the functional organisation of language
- surface forms are viewed as the consequences of selecting a set of abstract functional features
- choices correspond to minimal grammatical alternatives
- the interpolation of an intermediate abstract representation allows the specification of the text to accumulate gradually

173

Systemic Grammar



174

Systemic Grammar

Clause Choices

- Major indicative declarative:
  The cat is on the mat.
- Major indicative declarative relative:
  [He didn't see the cat] that chased the rat.
- Major indicative declarative bound:
  [It only hurts] when I laugh.
- Major indicative interrogative polar:
  Has anybody seen my seagull?
- Major imperative:
  Don't be ridiculous.
- Minor present-participle:
  [You'll enjoy] having more free time.

175

KPML

How it works:

- choices are made using INQUIRY SEMANTICS
- for each choice system in the grammar, a set of predicates known as CHOOSERS are defined
- these tests are functions from the internal state of the realiser and host generation system to one of the features in the system the chooser is associated with

176

KPML

Realisation Statements:

- small grammatical constraints at each choice point build up to a grammatical specification
- ⟨Insert SUBJECT⟩: an element functioning as subject will be present
- ⟨Conflate SUBJECT ACTOR⟩: the constituent functioning as SUBJECT is the same as the constituent that functions as ACTOR
- ⟨Order FINITE SUBJECT⟩: FINITE must immediately precede SUBJECT

177

Inputs and Outputs

```
(S1/ThereBe
   :object (O1/train
            :cardinality 20
            :relations
              ((R1/period :value daily)
               (R2/source :value Aberdeen)
               (R3/destination :value Glasgow))))
```

There are 20 trains daily from Aberdeen to Glasgow.

178

FUF/SURGE

- FUF: a unification-based linguistic realisation toolkit
- SURGE: a systemic-based unification grammar of English

179

FUF/SURGE

Basic idea:

- input specification in the form of a FUNCTIONAL DESCRIPTION, a recursive attribute--value matrix
- the grammar is a large functional description with alternations representing choice points
- realisation is achieved by unifying the input FD with the grammar FD

180

FUF: An Input FD

```
((cat clause)
 (process ((type composite)
           (relation possessive)
           (lex 'hand')))
 (participants ((agent ((cat pers_pro)
                        (gender feminine)))
                ((affected ❶((cat np)
                             (lex 'editor')))
                ((possessor ❶))
                ((possessed ((cat np)
                             (lex 'draft')))))
```

She hands the draft to the editor.

181

FUF: A Grammar Fragment

```
((cat np)
 (n ((cat noun)
     (number {^ ^ number})))
 (alt (
       ;; Proper names don't need an article
       ((proper yes)
        (pattern (n)))
       ;; Common names do
       ((proper no)
        (pattern (det n))
        (det ((cat article)
              (lex "the")))))))
```

182

Templates

- No grammar at all: instead, the NLG system is based on templates
  - Text Planner produces trees with template-like leaves
  - Sentence Planner performs usual operations
  - Realisation may do some morphology and orthography, but nothing else
- Text planning and sentence planning may still be very complex

183

An Template-driven Example

Text Planner Output:

```
[X  depart(num=num(X))   at  Y]
  X = CalExpress, Y = 1000
```

Sentence Planner Output:

```
[X  depart(num=num(X))   at  Y]
  X = [the Caledonian Express] (num=singular)
  Y = [10AM]
```

Realiser Output:

```
  The Caledonian Express departs at 10AM.
```

184

Templates: Pros and Cons

Pros

- easy for non-NLP people, such as domain experts, to understand
- easy and fast to implement

Cons

- lots of templates required if there is a lot of syntactic variability
- restricts sentence-planning possibilities

185

Morphology and Orthography

Realiser must be able to:

- inflect words
- apply standard orthographic spelling changes
- add punctuation
- add standard punctuation rules

186

Research Questions

- How can the different techniques be combined?
- How much sentence planning can be done with templates?
- How do layout issues affect realisation?

187

Linguistic Realization

- For the WeatherReporter system, complexity of the realiser depends on which linguistic decisions were already made by the sentence planner
- Simplest realiser just walks around the sentence plans in the text plan in a top-down left-to-right manner, realising leaf nodes
- More sophistication is possible

188

An SPL input to KPML

```
(l / greater-than-comparison
  :tense past
  :exceed-q (l  a) exceed
  :command-offer-q notcommandoffer
  :proposal-q notproposal
  :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.
```

189

An input FD for SURGE

```
((cat clause)
 (proc ((type ascriptive)
        (mode attributive)))
 (partic ((carrier ((cat common)
                    (lex "month")))
          (attribute ((cat ap)
                      (complex conjunction)
                      (distinct ~(((lex "cool")
                                   (comparative yes))
                                  ((lex "dry")
                                   (comparative yes))))))))))
```

The month was cooler and drier.

190

Overview

1 An Introduction to NLG
2 Requirements Analysis for NLG
3 NLG Architecture and System Design
4 A Case Study
5 A Closer Look at the Component Tasks
6 Conclusions and Pointers

191

Summary

We've seen

- techniques for text planning
- techniques for sentence planning
- techniques for linguistic realisation

192

Applied NLG in 1997

- Huge increase in the number of NLG applications being investigated or developed
- Slower increase in the number of applications which are actually fielded
  - but at least there are some: in 1987 there were none
- We are beginning to see reusable software, and specialist software houses
- Applied NLG is young, but the future looks bright

193

Resources: SIGGEN

SIGGEN (ACL Special Interest Group for Generation)

- Mailing list – contact siggen@cs.bgu.ac.il
- Web site at http://www.cs.bgu.ac.il/siggen
  - papers and bibliographies
  - conference and workshop announcements
  - software
  - job announcements
  - related pages

194

Resources: Conferences and Workshops

- International Workshop on NLG: every two years
- European Workshop on NLG: every two years, alternating with the International Workshops
- NLG papers at ACL, ANLP, IJCAI, AAAI ...
- See SIGGEN Web page for announcements

195

Resources: Papers

NLG Papers:

- The SIGGEN Web page:
  http://www.cs.bgu.ac.il/siggen
- The Computation and Language archive:
  http://xxx.lanl.gov/archive/cmp-lg

196

Resources: Software

- Realisers:
  - FUF
    http://www.cs.bgu.ac.il/fuf/
  - KPML
    http://www.darmstadt.gmd.de/publish/komet/kpml.html

197

Resources: Companies

Some software groups with NLG experience:

- CoGenTex: see http://www.cogentex.com
  - FoG system
  - several papers on new systems at ANLP97
- ERLI: see http://www.erli.com
  - AlethGen system
- Bellcore (SAIC): contact kukich@bellcore.com
  - PlanDoc system

198

Resources: Collections

G Kempen (ed) [1987] *Natural Language Generation*. Martinus Nijhoff Publishers.

D D McDonald and L Bolc (eds) [1988] *Natural Language Generation Systems*. Springer–Verlag.

M Zock and G Sabah (eds) [1988] *Advances in Natural Language Generation: An Interdisciplinary Perspective*. Two volumes. Pinter Publishers.

R Dale, C Mellish and M Zock [1990] *Current Research in Natural Language Generation*. Academic Press.

C L Paris, W R Swartout and W C Mann (eds) [1991] *Natural Language Generation in Artificial Intelligence and Computational Linguistics*. Kluwer Academic Publishers.

R Dale, E Hovy, D Rosner and O Stock (eds) [1992] *Aspects of Automated Natural Language Generation*. Springer–Verlag.

H Horacek and M Zock [1993] *New Concepts in Natural Language Generation*. Pinter Publishers.

G Adorni and M Zock (eds) [1996] *Trends in Natural Language Generation*. Springer–Verlag.

199

Resources: Books

Currently being written:

E. Reiter and R. Dale (1998)
Building Applied Text Generation Systems
Cambridge University Press

Don't miss it!

200