

NLG: Specific Components

Scott Farrar
CLMA, University of Washington
farrar@u.washington.edu

March 8, 2010

Today's lecture

- 1 Texts
- 2 NLG Systems
- 3 Architecture modules
 - Textplanner
 - Microplanner
 - Surface realizer
 - SimpleNLG realizer
- 4 Hw7

Example text (genealogy)

George Melvin Phillips (parents William D. Phillips and Matilda A. Jackson) was born 1864 in Athol, Somerset Co., MD. He died 14 April, 1933 in Allen, Wicomico Co., MD. Lillian White (parents George Melvin Phillips and Emma Washington Huffington) was born 27 October, 1908 in Allen, Wicomico Co., MD. She died 23 March, 1983 in Allen, Wicomico Co., MD.

What are the units of information expressed this text? What would need to be in a database in order to generate it?

Example text (genealogy)

George Melvin Phillips (*parents William D. Phillips and Matilda A. Jackson*) was born 1864 in Athol, Somerset Co., MD. He died 14 April, 1933 in Allen, Wicomico Co., MD. **Lillian White** (*parents George Melvin Phillips and Emma Washington Huffington*) was born 27 October, 1908 in Allen, Wicomico Co., MD. She died 23 March, 1983 in Allen, Wicomico Co., MD.

What are the units of information expressed this text? What would need to be in a database in order to generate it?

Example text (genealogy)

*George Melvin Phillips (parents **William D. Phillips and Matilda A. Jackson**) was born 1864 in Athol, Somerset Co., MD. He died 14 April, 1933 in Allen, Wicomico Co., MD. Lillian White (parents George Melvin Phillips and Emma Washington Huffington) was born 27 October, 1908 in Allen, Wicomico Co., MD. She died 23 March, 1983 in Allen, Wicomico Co., MD.*

What are the units of information expressed this text? What would need to be in a database in order to generate it?

Example text (genealogy)

*George Melvin Phillips (parents William D. Phillips and Matilda A. Jackson) was **born 1864** in Athol, Somerset Co., MD. He died 14 April, 1933 in Allen, Wicomico Co., MD. Lillian White (parents George Melvin Phillips and Emma Washington Huffington) was born 27 October, 1908 in Allen, Wicomico Co., MD. She died 23 March, 1983 in Allen, Wicomico Co., MD.*

What are the units of information expressed this text? What would need to be in a database in order to generate it?

Example text (genealogy)

George Melvin Phillips (parents William D. Phillips and Matilda A. Jackson) was born 1864 in Athol, Somerset Co., MD. He died 14 April, 1933 in Allen, Wicomico Co., MD. Lillian White (parents George Melvin Phillips and Emma Washington Huffington) was born 27 October, 1908 in Allen, Wicomico Co., MD. She died 23 March, 1983 in Allen, Wicomico Co., MD.

What are the units of information expressed this text? What would need to be in a database in order to generate it?

Example text (genealogy)

*George Melvin Phillips (parents William D. Phillips and Matilda A. Jackson) was born 1864 in Athol, Somerset Co., MD. He **died 14 April, 1933** in Allen, Wicomico Co., MD. Lillian White (parents George Melvin Phillips and Emma Washington Huffington) was born 27 October, 1908 in Allen, Wicomico Co., MD. She died 23 March, 1983 in Allen, Wicomico Co., MD.*

What are the units of information expressed this text? What would need to be in a database in order to generate it?

Example text (genealogy)

*George Melvin Phillips (parents William D. Phillips and Matilda A. Jackson) was born 1864 in Athol, Somerset Co., MD. He died 14 April, 1933 in **Allen**, Wicomico Co., MD. Lillian White (parents George Melvin Phillips and Emma Washington Huffington) was born 27 October, 1908 in Allen, Wicomico Co., MD. She died 23 March, 1983 in Allen, Wicomico Co., MD.*

What are the units of information expressed this text? What would need to be in a database in order to generate it?

Example text (personal ads)

I am a single girl new to area who would like to meet someone to hang out with and get a taste of local flavor. I am well educated with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential.... Don't reply without picture, and be single—no married or attached guys please.

Example text (personal ads)

*I am a **single girl** new to area who would like to meet someone to hang out with and get a taste of local flavor. I am well educated with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential.... Don't reply without picture, and be single—no married or attached guys please.*

Example text (personal ads)

*I am a single girl **new to area** who would like to meet someone to hang out with and get a taste of local flavor. I am well educated with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential.... Don't reply without picture, and be single—no married or attached guys please.*

Example text (personal ads)

*I am a single girl new to area who would **like to meet someone to hang out with** and get a taste of local flavor. I am well educated with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential.... Don't reply without picture, and be single—no married or attached guys please.*

Example text (personal ads)

*I am a single girl new to area who would like to meet someone to hang out with and **get a taste of local flavor**. I am well educated with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential.... Don't reply without picture, and be single—no married or attached guys please.*

Example text (personal ads)

*I am a single girl new to area who would like to meet someone to hang out with and get a taste of local flavor. I am **well educated** with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential.... Don't reply without picture, and be single—no married or attached guys please.*

Example text (personal ads)

*I am a single girl new to area who would like to meet someone to hang out with and get a taste of local flavor. I am well educated with a **great career**...full-figured. I'm looking for mutual stimulating conversation with dating potential.... Don't reply without picture, and be single—no married or attached guys please.*

Example text (personal ads)

*I am a single girl new to area who would like to meet someone to hang out with and get a taste of local flavor. I am well educated with a great career...**full-figured**. I'm looking for mutual stimulating conversation with dating potential.... Don't reply without picture, and be single—no married or attached guys please.*

Example text (personal ads)

I am a single girl new to area who would like to meet someone to hang out with and get a taste of local flavor. I am well educated with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential....
Don't reply without picture, and be single—no married or attached guys please.

Example text (personal ads)

I am a single girl new to area who would like to meet someone to hang out with and get a taste of local flavor. I am well educated with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential....
Don't reply without picture, and be single—no married or attached guys please.

There's more than just facts to be reported. In NLG communicative intention is crucial.

Example text (personal ads)

I am a single girl new to area who would like to meet someone to hang out with and get a taste of local flavor. I am well educated with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential....

Don't reply without picture, and be single—no married or attached guys please.

There's more than just facts to be reported. In NLG communicative intention is crucial.

Example text (personal ads)

I am a single girl new to area who would like to meet someone to hang out with and get a taste of local flavor. I am well educated with a great career...full-figured. I'm looking for mutual stimulating conversation with dating potential....

Don't reply without picture, and be single—no married or attached guys please.

There's more than just facts to be reported. In NLG communicative intention is crucial.

From knowledge to language

An important first step in NLG concerns **planning** the information needed to produce natural sounding, coherent text.

Texts

NLG Systems

Architecture modules

- Textplanner
- Microplanner
- Surface realizer
- SimpleNLG realizer

Hw7

From knowledge to language

An important first step in NLG concerns **planning** the information needed to produce natural sounding, coherent text.

From a non-linguistic knowledge base, the system needs to identify the information of interest and combine it in a way consistent with the way humans package their **beliefs, desires, intentions** as language.

Texts

NLG Systems

Architecture modules

- Textplanner
- Microplanner
- Surface realizer
- SimpleNLG realizer

Hw7

Today's lecture

- 1 Texts
- 2 NLG Systems
- 3 Architecture modules
 - Textplanner
 - Microplanner
 - Surface realizer
 - SimpleNLG realizer
- 4 Hw7

NLG Three-step systems

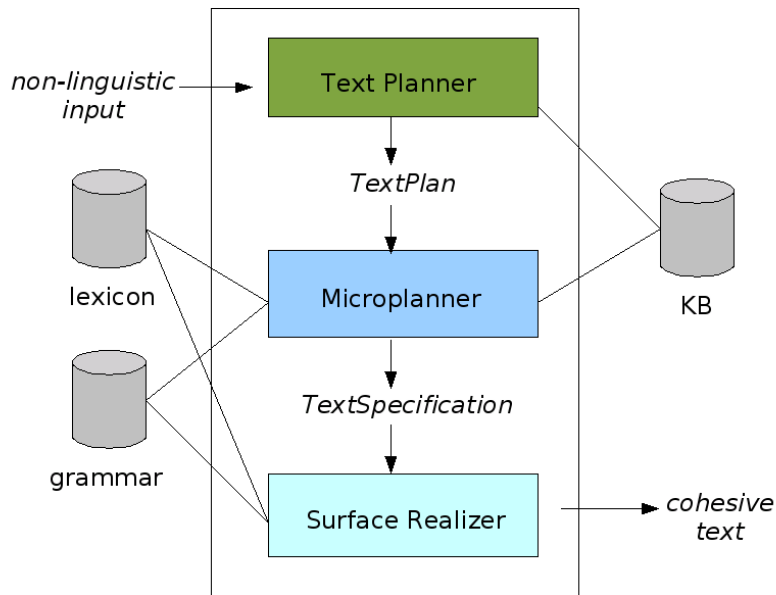
Last time we said that a more fine-tuned approach to the notion of **choice** was needed for better control over the NLG process. We compared a two- and a three-step approach.

Three-step architectures like WEATHERREPORTER or the KNIGHT System are more flexible, and modular. There's more control over the output.

Main components in a three-step system

Module	Content task	Structure task
<i>Text Planner</i>	content determination	document structuring
<i>Microplanner</i>	lexicalization; referring expression generation	aggregation
<i>Surface Realizer</i>	linguistic realization	structure realization

NLG three-step architecture



Today's lecture

- 1 Texts
- 2 NLG Systems
- 3 Architecture modules
 - Textplanner
 - Microplanner
 - Surface realizer
 - SimpleNLG realizer
- 4 Hw7

Text planner: Purpose

The text planner decides what chunks of information to include (content determination), and how to structure them (text structuring).

Text planner: Purpose

The text planner decides what chunks of information to include (content determination), and how to structure them (text structuring).

Definition

The basic unit of information produced by text planner is the **message**: a configuration of significant predications from the knowledge source. Messages correspond to major textual units in the output (e.g., a full sentence or group of sentences).

Text planner: Purpose

The text planner decides what chunks of information to include (content determination), and how to structure them (text structuring).

Definition

The basic unit of information produced by text planner is the **message**: a configuration of significant predications from the knowledge source. Messages correspond to major textual units in the output (e.g., a full sentence or group of sentences).

The text planner deals with information, not yet packaged in a linguistically suitable format, and with no reference to a NL grammar or lexicon.

Message types (genealogy vs. personal ads)

- MarriageMessage
- BirthMessage
- OccupationMessage

- PhysicalTraitMessage
- RelationshipStatusMessage
- LikesMessage, DislikesMessage
- OccupationMessage

Text planner: Input

- **knowledge source:** instances, classes, relations (possibly expressed in FOL, in a relational database, etc.)
- **communicative goals:** these are domain specific, though there are generalizations for all domains. E.g.,:
 - (*comparePerson Sam Fred*)
 - (*describeAll KB*)
 - (*queryAge Fred*)

Text planner: Output

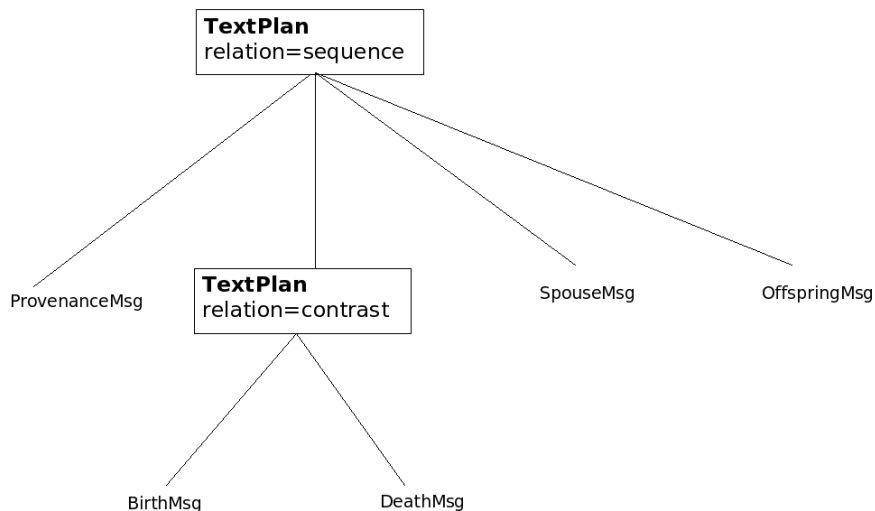
The text planner outputs a **text plan**: a non-linguistic data structure that contains messages structured according to rhetorical relations. It does not specify any grammatical or lexical information.

- **message**: *significant predications* from the domain.
E.g.
 $\exists e \text{ BirthEvent}(e) \wedge \text{actor}(e, \text{GEORGE}) \wedge \text{location}(e, \text{SOMERSET_CO_MD}), \text{ etc.}$
- **rhetorical structure**: relations between the chunks.
E.g,
elaboration, contrast, purpose, etc.

Target text elements

- George Melvin Phillips was from Somerset Co., MD.
- George Melvin Phillips was born in 1864.
- But George Melvin Phillips died in 1933.
- He was married to Martha Hastings.
- George and Martha had four children.

Output of text planner: *text plan*



Rhetorical Structure Theory (RST)

RST is a theory of the structure of texts that emphasizes textual function and the relationships between textual units. Consider several relation types among textual units $T1$ and $T2$:

Rhetorical Structure Theory (RST)

RST is a theory of the structure of texts that emphasizes textual function and the relationships between textual units. Consider several relation types among textual units $T1$ and $T2$:

- **evidence** - $T1$ is proven by $T2$.
The defendant killed Smith. He had Smith's blood on his hands.

Rhetorical Structure Theory (RST)

RST is a theory of the structure of texts that emphasizes textual function and the relationships between textual units. Consider several relation types among textual units $T1$ and $T2$:

- **evidence** - $T1$ is proven by $T2$.
The defendant killed Smith. He had Smith's blood on his hands.
- **elaboration** - the content of $T1$ is elaborated in $T2$.
Mary had a little lamb, and she had it with mint sauce.

Rhetorical Structure Theory (RST)

RST is a theory of the structure of texts that emphasizes textual function and the relationships between textual units. Consider several relation types among textual units $T1$ and $T2$:

- **evidence** - $T1$ is proven by $T2$.
The defendant killed Smith. He had Smith's blood on his hands.
- **elaboration** - the content of $T1$ is elaborated in $T2$.
Mary had a little lamb, and she had it with mint sauce.
- **sequence** - $T1$ precedes $T2$ in the narrative.
John picked up his iPhone, then fired his boss with a text message.

Rhetorical Structure Theory (RST)

RST is a theory of the structure of texts that emphasizes textual function and the relationships between textual units. Consider several relation types among textual units $T1$ and $T2$:

- **evidence** - $T1$ is proven by $T2$.
The defendant killed Smith. He had Smith's blood on his hands.
- **elaboration** - the content of $T1$ is elaborated in $T2$.
Mary had a little lamb, and she had it with mint sauce.
- **sequence** - $T1$ precedes $T2$ in the narrative.
John picked up his iPhone, then fired his boss with a text message.
- **contrast** - shows that two elements $T1$ and $T2$ are contrasting with each other.
The bride's dress was red, while the bridesmaids' were white.

Final structured text

George Melvin Phillips was from Somerset Co., MD. He was born in 1864, but died in 1933. He was married to Martha Hastings. Later, George and Martha had four children ...

Significant predications

Significant predications refer to the important knowledge to be linguistically packaged in the generated text. These must be determined in a domain specific manner, i.e., according to communicative goals.

Genealogy, encyclopedia entry, obituary, etc.

Roosevelt died of a cerebral hemorrhage on April 12, 1945

Significant predications

Significant predications refer to the important knowledge to be linguistically packaged in the generated text. These must be determined in a domain specific manner, i.e., according to communicative goals.

Genealogy, encyclopedia entry, obituary, etc.

Roosevelt died of a cerebral hemorrhage on April 12, 1945

Buddy Holley's coroner's report

Significant predications

Significant predications refer to the important knowledge to be linguistically packaged in the generated text. These must be determined in a domain specific manner, i.e., according to communicative goals.

Genealogy, encyclopedia entry, obituary, etc.

Roosevelt died of a cerebral hemorrhage on April 12, 1945

Buddy Holley's coroner's report

There was bleeding from both ears, and the face showed multiple lacerations. The consistency of the chest was soft due to extensive crushing injury to the bony structure. The left forearm was fractured 1/3 the way up from the wrist and the right elbow was fractured.

A BirthMessage from hw7

This structured object represents the information to generate:

FDR was born on January 30th, 1882 in Hyde Park, NY.

```
<BirthMessage>
  <person>
    <firstname>Franklin</firstname>
    <middlename>Delano</middlename>
    <lastname>Roosevelt</lastname>
    <gender>male</gender>
  </person>

  <date>
    <month>January</month>
    <day>30</day>
    <year>1882</year>
  </date>

  <location>
    <city>Hyde Park</city>
    <state>New York</state>
  </location>
</BirthMessage>
```

Challenges to two-level generation

Early generation systems (pre-1990): consisted of only two modules, were brittle and did not perform well in terms of generating the overall textual structure.

Challenges to two-level generation

Early generation systems (pre-1990): consisted of only two modules, were brittle and did not perform well in terms of generating the overall textual structure.

Challenges to two-level generation

Early generation systems (pre-1990): consisted of only two modules, were brittle and did not perform well in terms of generating the overall textual structure.

- 1 **Strategic generation:** to determine the significant predications and organize it into a text plan; research focused on AI planning techniques (e.g., STRIPS planner).

Challenges to two-level generation

Early generation systems (pre-1990): consisted of only two modules, were brittle and did not perform well in terms of generating the overall textual structure.

- 1 **Strategic generation:** to determine the significant predications and organize it into a text plan; research focused on AI planning techniques (e.g., STRIPS planner).
- 2 **Tactical generation:** grammatical selection; lexical choice (i.e., sentence planning)

Challenges to two-level generation

In early 1990s, researchers recognized that some functions of the strategic and tactical components overlapped:

Challenges to two-level generation

In early 1990s, researchers recognized that some functions of the strategic and tactical components overlapped:

- 1 The bridge between common-sense and linguistic knowledge is often taken to be the lexicon. There was much debate as to where exactly lexical selection belonged.

Challenges to two-level generation

In early 1990s, researchers recognized that some functions of the strategic and tactical components overlapped:

- 1 The bridge between common-sense and linguistic knowledge is often taken to be the lexicon. There was much debate as to where exactly lexical selection belonged.
- 2 In attempting complex text generation, the mapping of propositions/messages directly onto sentences resulted in choppy, robotic sounding texts. Combining information is especially difficult when the input data is not specially designed for NLG.

Challenges to two-level generation

In early 1990s, researchers recognized that some functions of the strategic and tactical components overlapped:

- 1 The bridge between common-sense and linguistic knowledge is often taken to be the lexicon. There was much debate as to where exactly lexical selection belonged.
- 2 In attempting complex text generation, the mapping of propositions/messages directly onto sentences resulted in choppy, robotic sounding texts. Combining information is especially difficult when the input data is not specially designed for NLG.
- 3 Consider the problem of how to refer to domain entities. In the best systems, control had to be switched back and forth between strategic and tactical components.

Referring expressions

AIG received another bailout ... The insurance giant underwent little scrutiny ... A spokesman for the company ... Congressmen berated the institution as ...

Two-step system: Control has to be switched back and forth between strategic and tactical components.

Referring expressions

AIG *received another bailout ... The insurance giant underwent little scrutiny ... A spokesman for the company ... Congressmen berated the institution as ...*

Two-step system: Control has to be switched back and forth between strategic and tactical components.

Referring expressions

AIG *received another bailout ... The insurance giant* **underwent little scrutiny ... A spokesman for the company ... Congressmen berated the institution as ...**

Two-step system: Control has to be switched back and forth between strategic and tactical components.

Referring expressions

AIG *received another bailout ... The insurance giant* *underwent little scrutiny ... A spokesman for the company ...* *Congressmen berated the institution as ...*

Two-step system: Control has to be switched back and forth between strategic and tactical components.

Referring expressions

AIG *received another bailout ...* The **insurance giant** *underwent little scrutiny ...* A *spokesman for the company ...* **Congressmen** *berated the institution as ...*

Two-step system: Control has to be switched back and forth between strategic and tactical components.

Microplanner: Purpose

These issues suggest the need for a third, intermediate component, often called a **microplanner**. This has been a major focus of research in the NLG community for the past decade or so.

Microplanner: Purpose

These issues suggest the need for a third, intermediate component, often called a **microplanner**. This has been a major focus of research in the NLG community for the past decade or so.

The microplanning component receives input from the text planner and determines the **deep linguistic** structure and content.

Microplanner: Purpose

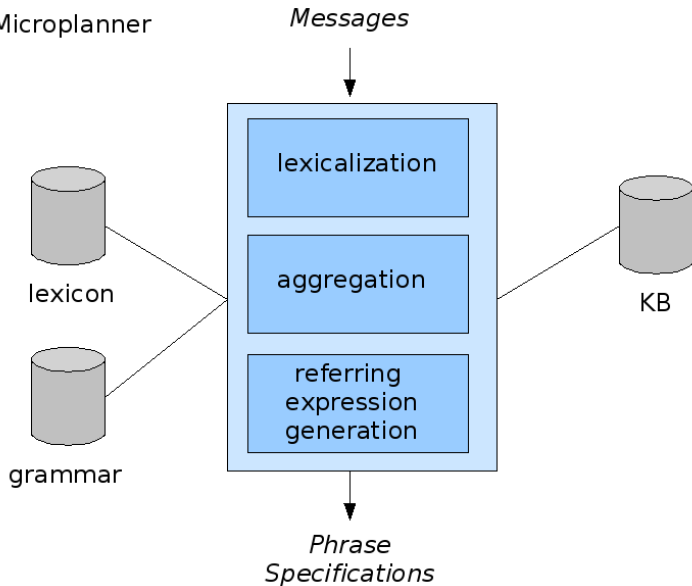
These issues suggest the need for a third, intermediate component, often called a **microplanner**. This has been a major focus of research in the NLG community for the past decade or so.

The microplanning component receives input from the text planner and determines the **deep linguistic** structure and content.

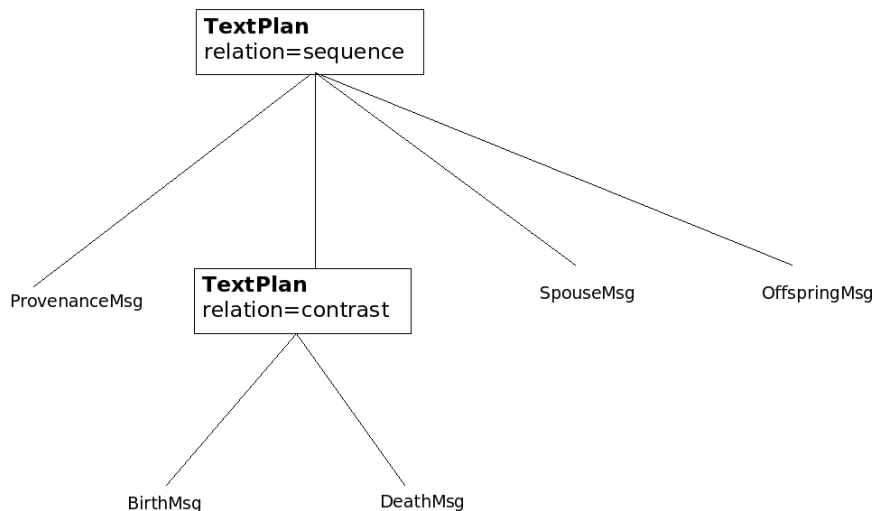
The main point: the microplanner is an intermediate stage that has access to both non-linguistic information (numerical database, etc.) and linguistic knowledge (grammar and lexicon).

Microplanner

Microplanner



Microplanner input



Microplanner output

NLG: Specific
Components

Scott Farrar
CLMA, University
of Washington far-
rar@u.washington.edu

Texts

NLG Systems

Architecture
modules

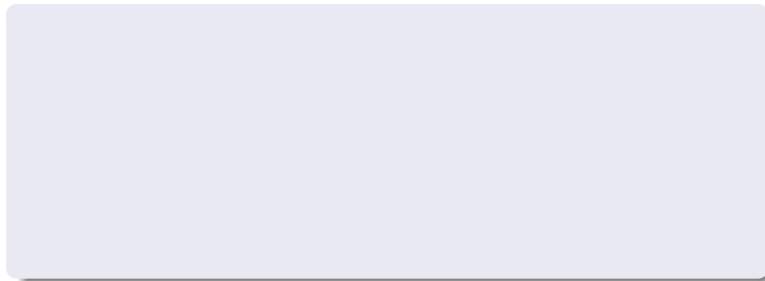
Textplanner

Microplanner

Surface realizer

SimpleNLG realizer

Hw7



Microplanner output

- a **phrase specification**: a data structure that, along with the grammar, gives a full recipe for a particular phrase (e.g., clause or noun phrase), but is not the phrase itself.

Microplanner output

- a **phrase specification**: a data structure that, along with the grammar, gives a full recipe for a particular phrase (e.g., clause or noun phrase), but is not the phrase itself.
- Or for generation of entire texts, a **text specification**, an abstract structure representing the text without committing to certain surface forms

Phrase specifications

Consider a phrase specification: we need two things in order to have a full recipe for the resulting NL:

Phrase specifications

Consider a phrase specification: we need two things in order
have a full recipe for the resulting NL:

Texts

NLG Systems

Architecture
modules

Textplanner
Microplanner
Surface realizer
SimpleNLG realizer

Hw7

Phrase specifications

Consider a phrase specification: we need two things in order have a full recipe for the resulting NL:

- 1 content: **lexemes** (linguistic counterparts of “concepts”)

Phrase specifications

Consider a phrase specification: we need two things in order have a full recipe for the resulting NL:

- 1 content: **lexemes** (linguistic counterparts of “concepts”)
- 2 structure:

Phrase specifications

Consider a phrase specification: we need two things in order have a full recipe for the resulting NL:

- 1 content: **lexemes** (linguistic counterparts of “concepts”)
- 2 structure:
 - phrasal categories,

Phrase specifications

Consider a phrase specification: we need two things in order have a full recipe for the resulting NL:

- 1 content: **lexemes** (linguistic counterparts of “concepts”)
- 2 structure:
 - phrasal categories,
 - features (whatever is grammaticalized in the language),

Phrase specifications

Consider a phrase specification: we need two things in order have a full recipe for the resulting NL:

- 1 content: **lexemes** (linguistic counterparts of “concepts”)
- 2 structure:
 - phrasal categories,
 - features (whatever is grammaticalized in the language),
 - semantic role information for mapping semantic to syntactic structure

Definition

A **lexeme** is an uninflected abstraction of a content word, e.g., *IDEA*, *SLEEP*, *GREEN* (cf. *ideas*, *slept*, *greener*). Dictionary entries are based on the idea of a lexeme.

Definition

A **lexeme** is an uninflected abstraction of a content word, e.g., *IDEA*, *SLEEP*, *GREEN* (cf. *ideas*, *slept*, *greener*). Dictionary entries are based on the idea of a lexeme.

The exact specification of a lexeme takes various forms for given theoretical traditions. But in general, lexemes are abstractions over a set of word forms.

Definition

The **lexicon** is the collection of lexemes for a given language. It provides a bridge from non-linguistic (common-sense) knowledge to linguistic knowledge (the grammar).

Definition

The **lexicon** is the collection of lexemes for a given language. It provides a bridge from non-linguistic (common-sense) knowledge to linguistic knowledge (the grammar).

The lexicon enumerates the psychologically and culturally salient concepts in a language.

Lexical semantics

A key aim of lexical semantics is to investigate the mapping between language and commonsense knowledge.

Lexical semantics

A key aim of lexical semantics is to investigate the mapping between language and commonsense knowledge.

Break or squash or crush?

John broke/squashed/crushed the ball.

Lexical semantics

A key aim of lexical semantics is to investigate the mapping between language and commonsense knowledge.

Break or squash or crush?

John broke/squashed/crushed the ball.

lexeme	semantic features
BREAK	object: +artifact, +rigid result: -functional

Lexical semantics

A key aim of lexical semantics is to investigate the mapping between language and commonsense knowledge.

Break or squash or crush?

John broke/squashed/crushed the ball.

lexeme	semantic features
BREAK	object: +artifact, +rigid result: -functional

Lexical semantics

A key aim of lexical semantics is to investigate the mapping between language and commonsense knowledge.

Break or squash or crush?

John broke/squashed/crushed the ball.

lexeme	semantic features
BREAK	object: +artifact, +rigid result: -functional
SQUASH	object: -rigid result: +flat, +smaller

Lexical semantics

A key aim of lexical semantics is to investigate the mapping between language and commonsense knowledge.

Break or squash or crush?

John broke/squashed/crushed the ball.

lexeme	semantic features
BREAK	object: +artifact, +rigid result: -functional
SQUASH	object: -rigid result: +flat, +smaller
CRUSH	object: +rigid result: +smaller

Structuring

Once the lexicalization is complete, the information in the message needs to be transformed into a grammatical form:

- the syntactic category (e.g., NP)
- the syntactic role (e.g., head of phrase, complement of phrase)
- the features relevant for the grammar (e.g., definiteness, tense, etc.)

Microplanning sub-tasks

In general, we can come up four subtasks for the microplanner:

- **grammaticalization**: committing to specific grammatical structures (NPs, VPs, features)
- **lexicalization**: committing to specific lexical items (lexemes for message content, cue words for textual relations)
- **aggregation**: repackaging message content into a form that is more language-like, and less data-like
- **referring expression generation**: generating specific forms for KB entities

Purpose

To generate natural language strings from a fully specified input (deterministic); the inverse of certain kinds of parsing processes.

- determines the surface form of the text;
- adds inflectional endings of words;
- orders constituents;
- misc. markup (e.g., lists, paragraphs, punctuation)

Texts

NLG Systems

Architecture
modules

Textplanner

Microplanner

Surface realizer

SimpleNLG realizer

Hw7

Surface realizer

Inputs

- phrase specifications
- Or for an entire text, a text specification

Outputs

- linearized sentences, texts

Details of the realizer

The surface realizer, in general, can be separated from the rest of the NLG system. It hides the idiosyncrasies of grammar from the rest of the system.

Details of the realizer

The surface realizer, in general, can be separated from the rest of the NLG system. It hides the idiosyncrasies of grammar from the rest of the system.

In theory, the output language (e.g., Spanish) could be changed by swapping out this component. It's the most *language specific* of the three components. (But this, really depends on how language-neutral the other NLG components are.)

Details of the realizer

The surface realizer, in general, can be separated from the rest of the NLG system. It hides the idiosyncrasies of grammar from the rest of the system.

In theory, the output language (e.g., Spanish) could be changed by swapping out this component. It's the most *language specific* of the three components. (But this, really depends on how language-neutral the other NLG components are.)

Cutting edge research does not focus on the realizer. Surface realization is largely a solved problem and there are a couple of robust open source systems.

Purpose

To take an underspecified input object (a text specification) and create a linearized string of words as output.

Texts

NLG Systems

Architecture
modules

Textplanner

Microplanner

Surface realizer

SimpleNLG realizer

Hw7

SimpleNLG

Purpose

To take an underspecified input object (a text specification) and create a linearized string of words as output.

Features of the system

Purpose

To take an underspecified input object (a text specification) and create a linearized string of words as output.

Features of the system

- programmatic lexicon access

Purpose

To take an underspecified input object (a text specification) and create a linearized string of words as output.

Features of the system

- programmatic lexicon access
- morphological component (e.g., adds *-s* to *dog*, *-ren* to *child*)

Purpose

To take an underspecified input object (a text specification) and create a linearized string of words as output.

Features of the system

- programmatic lexicon access
- morphological component (e.g., adds *-s* to *dog*, *-ren* to *child*)
- an inventory of morphological features

Purpose

To take an underspecified input object (a text specification) and create a linearized string of words as output.

Features of the system

- programmatic lexicon access
- morphological component (e.g., adds *-s* to *dog*, *-ren* to *child*)
- an inventory of morphological features
- various output formats: HTML, txt, etc.

Purpose

To take an underspecified input object (a text specification) and create a linearized string of words as output.

Features of the system

- programmatic lexicon access
- morphological component (e.g., adds *-s* to *dog*, *-ren* to *child*)
- an inventory of morphological features
- various output formats: HTML, txt, etc.
- structured objects representing text hierarchy

Elements

“Realisation in SimpleNLG revolves around a tree structure. Each node in the tree is represented by a NLGElement, which in turn may have child nodes.”

Direct subclasses of NLGElement

These are the primary elements:

Elements

“Realisation in SimpleNLG revolves around a tree structure. Each node in the tree is represented by a `NLGElement`, which in turn may have child nodes.”

Direct subclasses of `NLGElement`

These are the primary elements:

- `DocumentElement`: used to define elements that form part of the textual structure

“Realisation in SimpleNLG revolves around a tree structure. Each node in the tree is represented by a `NLGElement`, which in turn may have child nodes.”

Direct subclasses of `NLGElement`

These are the primary elements:

- `DocumentElement`: used to define elements that form part of the textual structure
- `PhraseElement`: defines a phrase and covers the expected phrase types: noun phrases, verb phrases, etc.

“Realisation in SimpleNLG revolves around a tree structure. Each node in the tree is represented by a `NLGElement`, which in turn may have child nodes.”

Direct subclasses of `NLGElement`

These are the primary elements:

- `DocumentElement`: used to define elements that form part of the textual structure
- `PhraseElement`: defines a phrase and covers the expected phrase types: noun phrases, verb phrases, etc.
- `WordElement`: the class for a lexical entry (ie, a word), stored in a `Lexicon`

Elements continued

Direct subclasses of NLGElement

Other elements:

Texts

NLG Systems

Architecture
modules

Textplanner

Microplanner

Surface realizer

SimpleNLG realizer

Hw7

Direct subclasses of NLGElement

Other elements:

- `CoordinatedPhraseElement`: defines coordination between two or more phrases and involves the use of key words such as *and* or *but*.

Texts

NLG Systems

Architecture
modules

Textplanner

Microplanner

Surface realizer

SimpleNLG realizer

Hw7

Direct subclasses of NLGElement

Other elements:

- `CoordinatedPhraseElement`: defines coordination between two or more phrases and involves the use of key words such as *and* or *but*.
- `InflectedWordElement`: used to represent an word that requires inflection by the morphology

Texts

NLG Systems

Architecture
modules

Textplanner

Microplanner

Surface realizer

SimpleNLG realizer

Hw7

Direct subclasses of NLGElement

Other elements:

- `CoordinatedPhraseElement`: defines coordination between two or more phrases and involves the use of key words such as *and* or *but*.
- `InflectedWordElement`: used to represent a word that requires inflection by the morphology
- `ListElement`: used to define elements that can be grouped together and treated in a similar manner

Texts

NLG Systems

Architecture
modules

Textplanner

Microplanner

Surface realizer

SimpleNLG realizer

Hw7

Direct subclasses of NLGElement

Other elements:

- `CoordinatedPhraseElement`: defines coordination between two or more phrases and involves the use of key words such as *and* or *but*.
- `InflectedWordElement`: used to represent a word that requires inflection by the morphology
- `ListElement`: used to define elements that can be grouped together and treated in a similar manner
- `StringElement`: an element for representing canned text

Texts

NLG Systems

Architecture
modules

Textplanner

Microplanner

Surface realizer

SimpleNLG realizer

Hw7

Canned text

Canned text refers to the use of set words or phrases in place of programmatic generation.

Examples:

Instead of generating a VP using phrasal elements and lexical item, simply use a string: “becoming partly cloudy”.

```
StringElement canned = new  
    StringElement("becoming partly cloudy");
```

Today's lecture

- 1 Texts
- 2 NLG Systems
- 3 Architecture modules
 - Textplanner
 - Microplanner
 - Surface realizer
 - SimpleNLG realizer
- 4 Hw7

Strategies for Hw7

NLG: Specific
Components

Scott Farrar
CLMA, University
of Washington far-
rar@u.washington.edu

Create various message types

For each message type in the XML input, create a class and then instantiate that class with the input. Message types will contain references to the semantic content.

Texts

NLG Systems

Architecture
modules

Textplanner
Microplanner
Surface realizer
SimpleNLG realizer

Hw7

Strategies for Hw7

Create various message types

For each message type in the XML input, create a class and then instantiate that class with the input. Message types will contain references to the semantic content.

Create various semantic classes

- **Person**: a class representing a person (name, gender, etc.)
- **Date**: a class representing a data (day, month, year)
- **Location**: a class representing a location (city, state)