Inducing Document Plans for Concept-to-Text Generation

Ioannis Konstas, Mirella Lapata

Institute for Language, Cognition and Computation
University of Edinburgh

EMNLP 2013, Seattle
20 October 2013

Introduction

Concept-to-text generation refers to the task of automatically producing textual output from nonlinguistic input (Reiter and Dale, 2000)

Introduction

Concept-to-text generation refers to the task of automatically producing textual output from nonlinguistic input (Reiter and Dale, 2000)

Desktop		
Cmd	**Name**	**Type**
left-click	start	button

Start		
Cmd	**Name**	**Type**
left-click	settings	button

Location		
Name		**Type**
start menu		button
control panel		window

Start Target		
Cmd	**Name**	**Type**
left-click	control panel	button

Navigate Window		
Cmd	**Name**	**Type**
left-click	accounts and users	window

Context Menu		
Cmd	**Name**	**Type**
left-click	advanced	tab

Action Context Menu		
Cmd	**Name**	**Type**
left-click	advanced	button

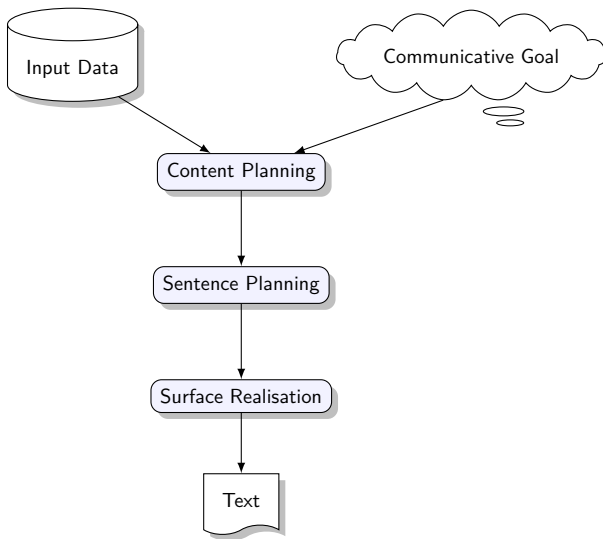| Window Target		
Cmd	**Name**	**Type**
double-click	users and passwords	item

Click start, point to settings, and then click control panel.
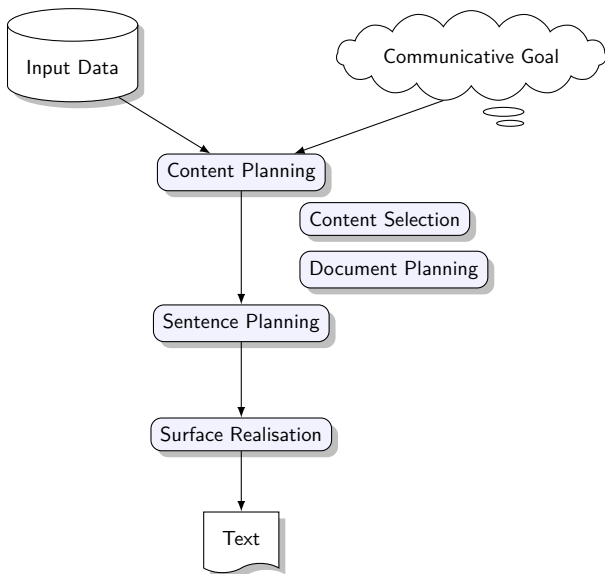
Double-click users and passwords.
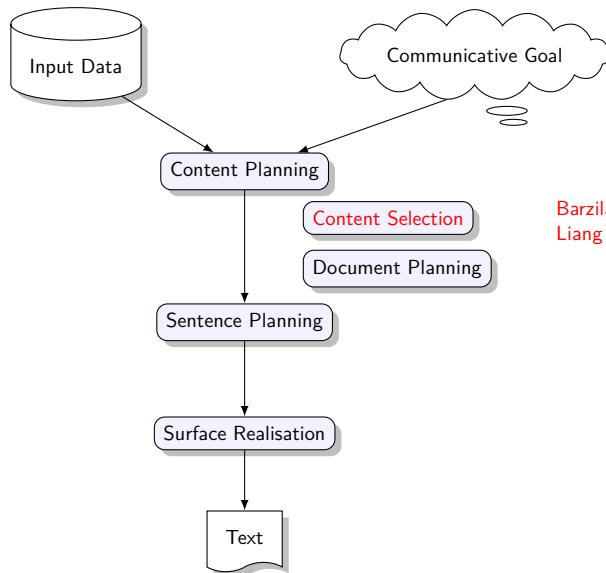
On the advanced tab, click advanced.

Traditional NLG Pipeline
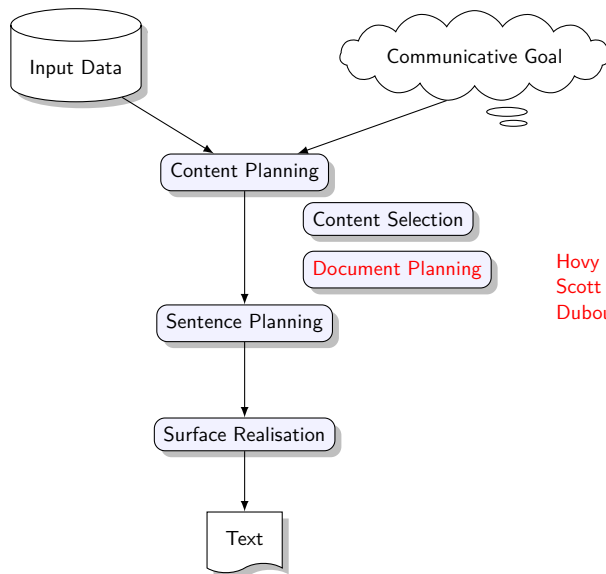
Traditional NLG Pipeline

Traditional NLG Pipeline

Input Data

Communicative Goal

Content Planning

Content Selection

Document Planning

Barzilay and Lapata (2005)
Liang et al. (2009)

Sentence Planning

Surface Realisation

Text

Traditional NLG Pipeline

Input Data

Communicative Goal

Content Planning

Content Selection

Document Planning

Hovy (1993)
Scott and de Souza (1990)
Duboue and Mckeown (2002)

Sentence Planning

Surface Realisation

Text

Traditional NLG Pipeline

Input Data

Communicative Goal

Content Planning

Content Selection

Document Planning

Sentence Planning

Stent et al. (2004)

Surface Realisation

Text

Traditional NLG Pipeline

Input Data

Communicative Goal

Content Planning

Content Selection

Document Planning

Sentence Planning

Surface Realisation
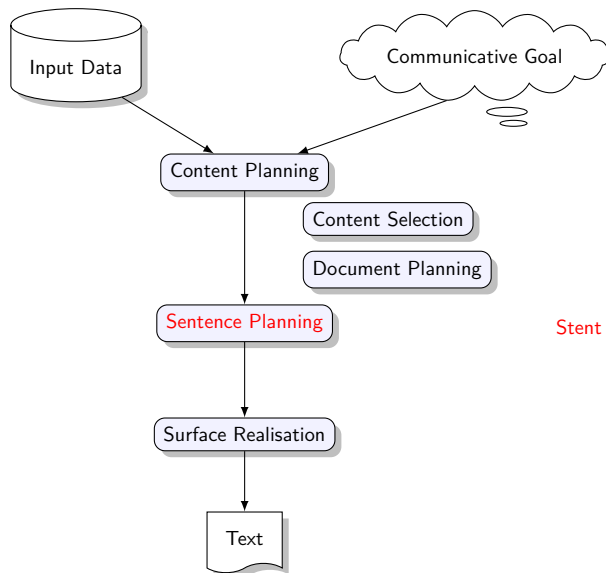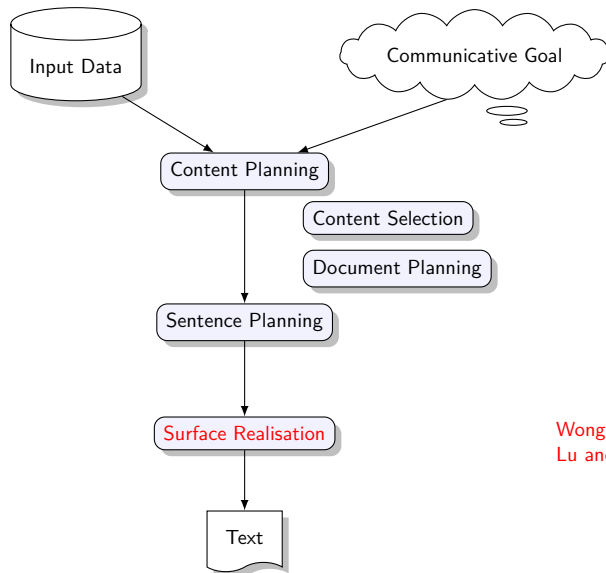
Wong and Mooney (2007)
Lu and Ng (2011)

Text

Traditional NLG Pipeline

Input Data

Communicative Goal

Content Planning

Content Selection

Document Planning

Kim and Mooney (2010)
Pipeline approach

Sentence Planning

Surface Realisation

Text

Traditional NLG Pipeline

Input Data

Communicative Goal

Content Planning

Content Selection

Kim and Mooney (2010)
Pipeline approach

Document Planning

Angeli et al. (2010)
Unified approach

Sentence Planning

Surface Realisation

Text

Traditional NLG Pipeline

Input Data

Communicative Goal

Content Planning

Content Selection

Document Planning

Sentence Planning

Surface Realisation

Text

Kim and Mooney (2010)
Pipeline approach

Angeli et al. (2010)
Unified approach

Konstas and Lapata (2012)
Joint approach

Traditional NLG Pipeline

Input Data

Communicative Goal

Content Planning

Content Selection

Document Planning

Sentence Planning

Surface Realisation

Text

Kim and Mooney (2010)
Pipeline approach

Angeli et al. (2010)
Unified approach

Konstas and Lapata (2012)
Joint approach

This work
Joint approach

Input

- Input: database records **d**
- Output: words **w** corresponding to some records of **d**
- Each record $r \in$ **d** has a type $r.t$ and fields f
- Fields have values $f.v$ and types $f.t$ (integer, categorical, string)

Desktop		
Cmd	**Name**	**Type**
left-click	start	button

click start,

Input

- Input: database records **d**
- Output: words **w** corresponding to some records of **d**
- Each record $r \in$ **d** has a type $r.t$ and fields f
- Fields have values $f.v$ and types $f.t$ (integer, categorical, string)

Desktop		
Cmd	**Name**	**Type**
left-click	start	button

click start,

Input

- Input: database records **d**
- Output: words **w** corresponding to some records of **d**
- Each record $r \in$ **d** has a type $r.t$ and fields f
- Fields have values $f.v$ and types $f.t$ (integer, categorical, string)

Desktop		
Cmd	**Name**	**Type**
left-click	start	button

click start,

Input

- Input: database records **d**
- Output: words **w** corresponding to some records of **d**
- Each record $r \in$ **d** has a type $r.t$ and fields f
- Fields have values $f.v$ and types $f.t$ (integer, categorical, string)

Desktop		
Cmd	**Name**	**Type**
left-click	start	button

click start,

Input

- Input: database records **d**
- Output: words **w** corresponding to some records of **d**
- Each record $r \in$ **d** has a type $r.t$ and fields f
- Fields have values $f.v$ and types $f.t$ (integer, categorical, string)

Desktop		
Cmd	**Name**	**Type**
left-click	start	button

click start,

Key Idea

Desktop		
Cmd	Name	Type
left-click	start	button

Start		
Cmd	Name	Type
left-click	settings	button

Location		
Name	Type
start menu	button
control panel	window

Start Target		
Cmd	Name	Type
left-click	control panel	button

Navigate Window		
Cmd	Name	Type
left-click	accounts and users	window

Context Menu		
Cmd	Name	Type
left-click	advanced	tab

Action Context Menu		
Cmd	Name	Type
left-click	advanced	button

Window Target		
Cmd	Name	Type
double-click	users and passwords	item

> Click start, point to settings, and then click control panel.
>
> Double-click users and passwords.
>
> On the advanced tab, click advanced.

Key Idea

Desktop		
Cmd	**Name**	**Type**
left-click	start	button

Start		
Cmd	**Name**	**Type**
left-click	settings	button

Location		
~~Name~~		~~Type~~
~~start menu~~		~~button~~
~~control panel~~		~~window~~

Start Target		
Cmd	**Name**	**Type**
left-click	control panel	button

Navigate Window		
~~Cmd~~	~~Name~~	~~Type~~
~~left-click~~	~~accounts and users~~	~~window~~

Context Menu		
Cmd	**Name**	**Type**
left-click	advanced	tab

Action Context Menu		
Cmd	**Name**	**Type**
left-click	advanced	button

Window Target		
Cmd	**Name**	**Type**
double-click	users and passwords	item

> Click start, point to settings, and then click control panel.
>
> Double-click users and passwords.
>
> On the advanced tab, click advanced.

Key Idea

Desktop		
Cmd	**Name**	**Type**
left-click	start	button

Start		
Cmd	**Name**	**Type**
left-click	settings	button

Start Target		
Cmd	**Name**	**Type**
left-click	control panel	button

Window Target		
Cmd	**Name**	**Type**
double-click	users and passwords	item

Context Menu		
Cmd	**Name**	**Type**
left-click	advanced	tab

Action Context Menu		
Cmd	**Name**	**Type**
left-click	advanced	button

> Click start, point to settings, and then click control panel.
>
> Double-click users and passwords.
>
> On the advanced tab, click advanced.
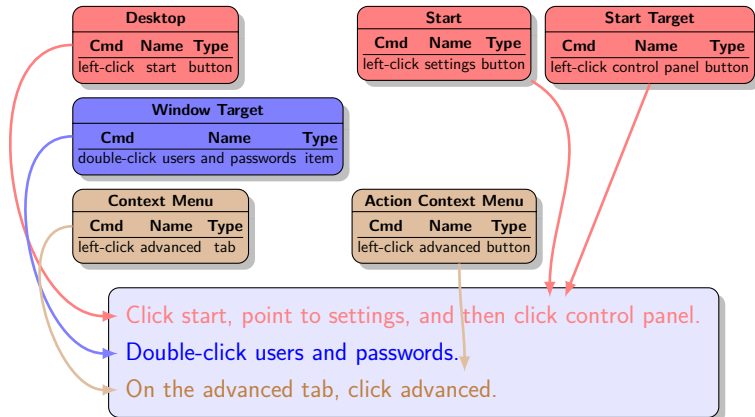
Key Idea

Key Idea
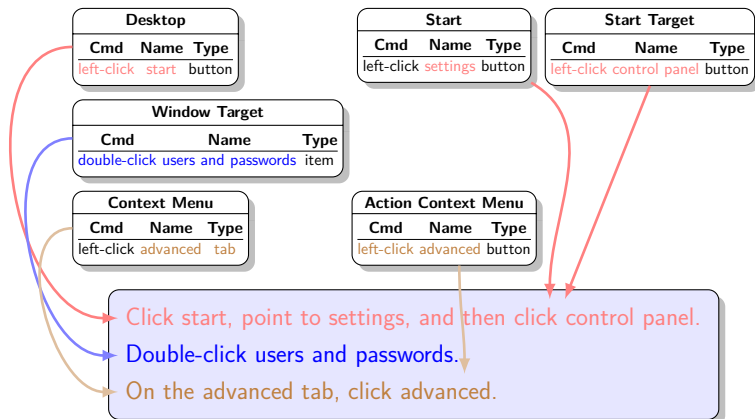
Key Idea

Key Idea: Grammar-based document plans

Key Idea

Key Idea: Grammar-based document plans

- Extend the model of Konstas and Lapata (2012)

Key Idea

Key Idea: Grammar-based document plans

- Extend the model of Konstas and Lapata (2012)
- Re-use the generation model based on a PCFG grammar of input

Key Idea

Key Idea: Grammar-based document plans

- Extend the model of Konstas and Lapata (2012)
- Re-use the generation model based on a PCFG grammar of input
- Replace existing locally coherent **Content Selection** model and incorporate global **Document Planning** (explore two solutions):

Key Idea

Key Idea: Grammar-based document plans

- Extend the model of Konstas and Lapata (2012)
- Re-use the generation model based on a PCFG grammar of input
- Replace existing locally coherent **Content Selection** model and incorporate global **Document Planning** (explore two solutions):

Key Idea

Key Idea: Grammar-based document plans

- Extend the model of Konstas and Lapata (2012)
- Re-use the generation model based on a PCFG grammar of input
- Replace existing locally coherent **Content Selection** model and incorporate global **Document Planning** (explore two solutions):

Patterns of record sequences *within* a sentence and *among* sentences

Rhetorical Structure Theory (Mann and Thompson, 1988) inspired plans
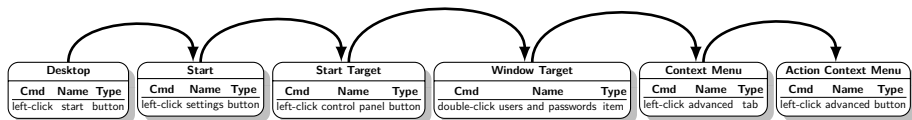
Grammar

Grammar

1. $S \rightarrow R(start)$

Grammar

1. $S \rightarrow R(start)$
2. $R(r_i.t) \rightarrow FS(r_j, start)R(r_j.t) \mid FS(r_j, start)$

$R(desktop_1.t) \rightarrow FS(start_1, start)R(start_1.t)$

Grammar

Desktop		
Cmd	**Name**	**Type**
left-click	start	button

Start		
Cmd	**Name**	**Type**
left-click	settings	button

Start Target		
Cmd	**Name**	**Type**
left-click	control panel	button

Window Target		
Cmd	**Name**	**Type**
double-click	users and passwords	item

Context Menu		
Cmd	**Name**	**Type**
left-click	advanced	tab

Action Context Menu		
Cmd	**Name**	**Type**
left-click	advanced	button

1. $S \rightarrow R(start)$
2. $R(r_i.t) \rightarrow FS(r_j, start)R(r_j.t) \mid FS(r_j, start)$

$R(desktop_1.t) \rightarrow FS(start_1, start)R(start_1.t)$

Grammar

Desktop		
Cmd	Name	Type
left-click	start	button

Start		
Cmd	Name	Type
left-click	settings	button

Start Target		
Cmd	Name	Type
left-click	control panel	button

Window Target		
Cmd	Name	Type
double-click	users and passwords	item

Context Menu		
Cmd	Name	Type
left-click	advanced	tab

Action Context Menu		
Cmd	Name	Type
left-click	advanced	button

1. $S \rightarrow R(start)$

2. $R(r_i.t) \rightarrow FS(r_j, start)R(r_j.t) \mid FS(r_j, start)$

3. $FS(r, r.f_i) \rightarrow F(r, r.f_j)FS(r, r.f_j) \mid F(r, r.f_j)$

$FS(desktop_1, cmd) \rightarrow F(desktop_1, name)FS(desktop_1, name)$

Grammar

Desktop				Start				Start Target				Window Target			Context Menu				Action Context Menu		
Cmd	Name	Type		Cmd	Name	Type		Cmd	Name	Type		Cmd	Name	Type	Cmd	Name	Type		Cmd	Name	Type
left-click	start	button		left-click	settings	button		left-click	control panel	button		double-click	users and passwords	item	left-click	advanced	tab		left-click	advanced	button
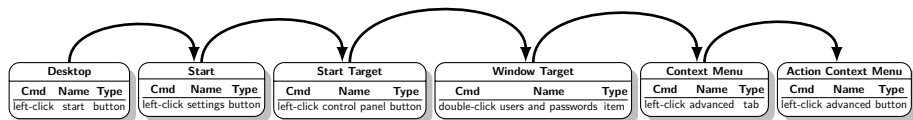
1. $S \rightarrow R(start)$

2. $R(r_i.t) \rightarrow FS(r_j, start)R(r_j.t) \mid FS(r_j, start)$

3. $FS(r, r.f_i) \rightarrow F(r, r.f_j)FS(r, r.f_j) \mid F(r, r.f_j)$

4. $F(r, r.f) \rightarrow W(r, r.f)F(r, r.f) \mid W(r, r.f)$

$F(desktop_1, cmd) \rightarrow W(desktop_1, cmd)F(desktop_1, cmd)$

Grammar
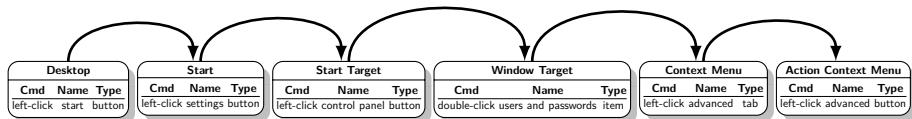
1. $S \rightarrow R(start)$
2. $R(r_i.t) \rightarrow FS(r_j, start)R(r_j.t) \mid FS(r_j, start)$
3. $FS(r, r.f_i) \rightarrow F(r, r.f_j)FS(r, r.f_j) \mid F(r, r.f_j)$
4. $F(r, r.f) \rightarrow W(r, r.f)F(r, r.f) \mid W(r, r.f)$
5. $W(r, r.f) \rightarrow \alpha \mid g(f.v)$

$W(desktop_1, cmd) \rightarrow$ click $[cmd.v = \text{`left-click'}]$

Grammar

1. $S \rightarrow R(start)$
2. $R(r_i.t) \rightarrow FS(r_j, start)R(r_j.t) \mid FS(r_j, start)$
3. $FS(r, r.f_i) \rightarrow F(r, r.f_j)FS(r, r.f_j) \mid F(r, r.f_j)$
4. $F(r, r.f) \rightarrow W(r, r.f)F(r, r.f) \mid W(r, r.f)$
5. $W(r, r.f) \rightarrow \alpha \mid g(f.v)$

EM Training: dynamic program similar to the inside-outside algorithm

Planning with Record Sequences

Key idea: Grammar on sequences of record types (G_{RSE})

Planning with Record Sequences

Key idea: Grammar on sequences of record types (G_{RSE})

1. Click start, point to settings, and then click control panel. ‖ Double-click users and passwords. ‖ On the advanced tab, click advanced. ‖
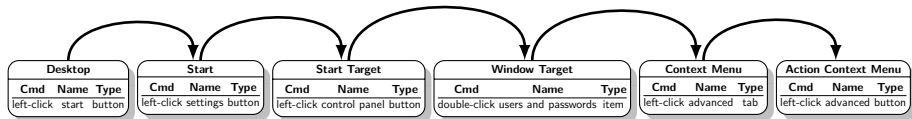
   Split a document into sentences, each terminated by a full-stop.

Planning with Record Sequences

Key idea: Grammar on sequences of record types (G_{RSE})

1. Click start, point to settings, and then click control panel. ‖ Double-click users and passwords. ‖ On the advanced tab, click advanced. ‖

   Split a document into sentences, each terminated by a full-stop.

2. $\dfrac{\text{desktop | start | start-target}}{\text{Click start, point to settings, and then click control panel.}}$ ‖
   $\dfrac{\text{window-target}}{\text{Double-click users and passwords.}}$ ‖ $\dfrac{\text{contextMenu | action-contextMenu}}{\text{On the advanced tab, click advanced.}}$ ‖

   Then split a sentence further into a sequence of record types.

# Planning with Record Sequences

Key idea: Grammar on sequences of record types (G_{RSE})

1. Click start, point to settings, and then click control panel. ‖ Double-click users and passwords. ‖ On the advanced tab, click advanced. ‖

   Split a document into sentences, each terminated by a full-stop.

2. $$\frac{\text{desktop} \mid \text{start} \mid \text{start-target}}{\text{Click start, point to settings, and then click control panel.}} \; \| \; \frac{\text{window-target}}{\text{Double-click users and passwords.}} \; \| \; \frac{\text{contextMenu} \mid \text{action-contextMenu}}{\text{On the advanced tab, click advanced.}} \; \|$$

   Then split a sentence further into a sequence of record types.

3. Goal: Learn patterns of record type sequences within and among sentences

Extended Grammar

Desktop		
Cmd	Name	Type
left-click	start	button

Start		
Cmd	Name	Type
left-click	settings	button

Start Target		
Cmd	Name	Type
left-click	control panel	button

Window Target		
Cmd	Name	Type
double-click	users and passwords	item

Context Menu		
Cmd	Name	Type
left-click	advanced	tab

Action Context Menu		
Cmd	Name	Type
left-click	advanced	button

1. $S \rightarrow R(\textit{start})$
2. $R(r_i.t) \rightarrow FS(r_j, \textit{start})R(r_j.t) \mid FS(r_j, \textit{start})$
3. $FS(r, r.f_i) \rightarrow F(r, r.f_j)FS(r, r.f_j) \mid F(r, r.f_j)$
4. $F(r, r.f) \rightarrow W(r, r.f)F(r, r.f) \mid W(r, r.f)$
5. $W(r, r.f) \rightarrow \alpha \mid g(f.v)$

Extended Grammar
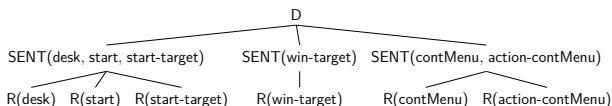
1. $D \rightarrow SENT(t_i, \ldots, t_j) \ldots SENT(t_l, \ldots, t_m)$

2. $SENT(t_i, \ldots, t_j) \rightarrow R(r_a.t_i) \ldots R(r_k.t_j) \cdot$

3. $R(r_i.t) \rightarrow FS(r_j, start)$

4. $FS(r, r.f_i) \rightarrow F(r, r.f_j)FS(r, r.f_j) \mid F(r, r.f_j)$

5. $F(r, r.f) \rightarrow W(r, r.f)F(r, r.f) \mid W(r, r.f)$

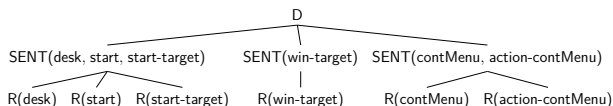6. $W(r, r.f) \rightarrow \alpha \mid g(f.v) \mid gen\_str(f.v, i)$

Extended Grammar
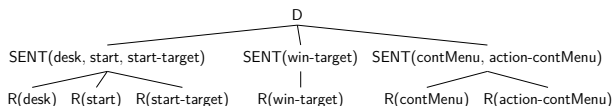
1. $D \rightarrow SENT(t_i, \ldots, t_j) \ldots SENT(t_l, \ldots, t_m)$
2. $SENT(t_i, \ldots, t_j) \rightarrow R(r_a.t_i) \ldots R(r_k.t_j) \cdot$
3. $R(r_i.t) \rightarrow FS(r_j, start)$
4. $FS(r, r.f_i) \rightarrow F(r, r.f_j)FS(r, r.f_j) \mid F(r, r.f_j)$
5. $F(r, r.f) \rightarrow W(r, r.f)F(r, r.f) \mid W(r, r.f)$
6. $W(r, r.f) \rightarrow \alpha \mid g(f.v) \mid gen\_str(f.v, i)$

Straightforward solution: Embed the parameters with the original grammar and train using EM

Extended Grammar

1. $D \rightarrow SENT(t_i, \ldots, t_j) \ldots SENT(t_l, \ldots, t_m)$
2. $SENT(t_i, \ldots, t_j) \rightarrow R(r_a.t_i) \ldots R(r_k.t_j) \cdot$
3. $R(r_i.t) \rightarrow FS(r_j, start)$
4. $FS(r, r.f_i) \rightarrow F(r, r.f_j)FS(r, r.f_j) \mid F(r, r.f_j)$
5. $F(r, r.f) \rightarrow W(r, r.f)F(r, r.f) \mid W(r, r.f)$
6. $W(r, r.f) \rightarrow \alpha \mid g(f.v) \mid gen\_str(f.v, i)$

Straightforward solution: Embed the parameters with the original grammar and train using EM

Plan B: Extract grammar rules from training data

Grammar Extraction

desktop	start	start-target	window-target
Click start,	point to settings,	and then click control panel.	Double-click users and passwords.

contextMenu	action-contextMenu
On the advanced tab ,	click advanced.

Liang et al. (2009)

Grammar Extraction

desktop	start	start-target	window-target
Click start,	point to settings,	and then click control panel.	Double-click users and passwords.

contextMenu	action-contextMenu
On the advanced tab ,	click advanced.

Liang et al. (2009)

⇩

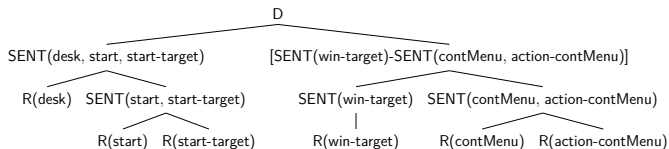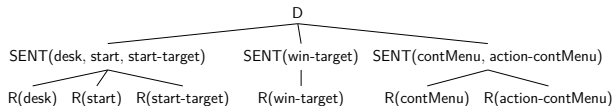$\Big[$ desktop start start-target ‖ window-target ‖ contextMenu action-contMenu ‖ $\Big]$

Grammar Extraction

desktop	start	start-target	window-target
Click start,	point to settings,	and then click control panel.	Double-click users and passwords.

contextMenu	action-contextMenu
On the advanced tab ,	click advanced.

Liang et al. (2009)

$$\Downarrow$$
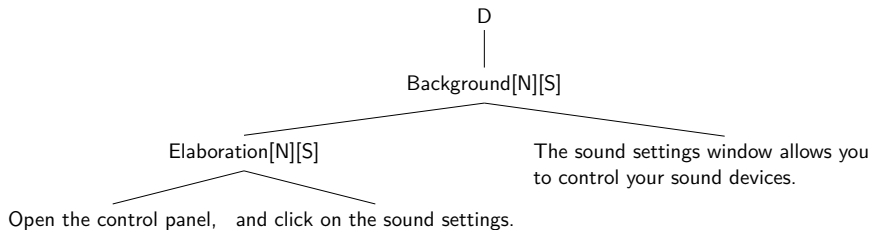
$$\left[\text{desktop start start-target} \parallel \text{window-target} \parallel \text{contextMenu action-contMenu} \parallel \right]$$
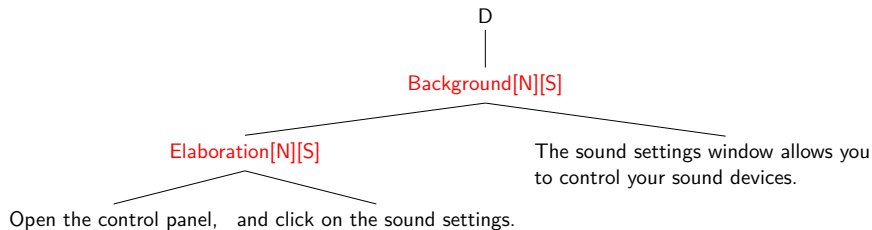
```
                                    D
        ┌───────────────┬───────────────┴────────────────┐
SENT(desk, start, start-target)  SENT(win-target)  SENT(contMenu, action-contMenu)
    ┌──────┼──────┐              │              ┌──────────┴────────┐
R(desk) R(start) R(start-target)  R(win-target)  R(contMenu)  R(action-contMenu)
```

Grammar Extraction

desktop	start	start-target	window-target
Click start,	point to settings,	and then click control panel.	Double-click users and passwords.

contextMenu	action-contextMenu
On the advanced tab ,	click advanced.

Liang et al. (2009)

$$\Downarrow$$

$\left[\text{desktop start start-target} \parallel \text{window-target} \parallel \text{contextMenu action-contMenu} \parallel \right]$

```
                              D
        ┌─────────────────────┼─────────────────────┐
SENT(desk, start, start-target)  SENT(win-target)  SENT(contMenu, action-contMenu)
    ┌────┬────┬────┐              │              ┌───────────┴───────────┐
R(desk) R(start) R(start-target)  R(win-target)  R(contMenu)  R(action-contMenu)
```

```
                                        D
           ┌────────────────────────────┴────────────────────────────┐
  SENT(desk, start, start-target)      [SENT(win-target)-SENT(contMenu, action-contMenu)]
     ┌──────┴──────┐                    ┌───────────────┴───────────────┐
  R(desk)  SENT(start, start-target)   SENT(win-target)   SENT(contMenu, action-contMenu)
           ┌──────┴──────┐                   │              ┌───────────┴───────────┐
      R(start)  R(start-target)         R(win-target)   R(contMenu)  R(action-contMenu)
```

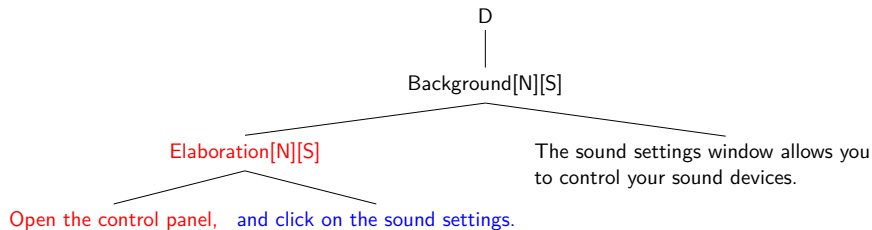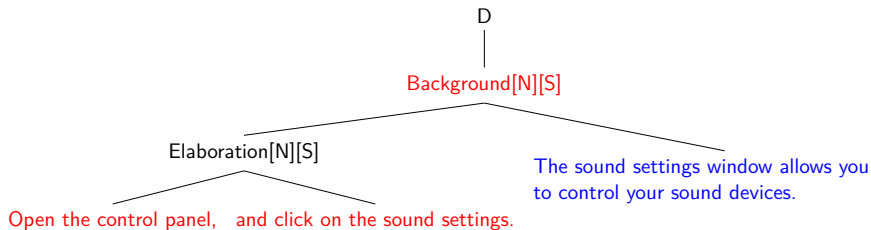# Planning with Rhetorical Structure Theory

RST (Mann and Thompson, 1988)

```
                              D
                              |
                      Background[N][S]
                   /                        \
          Elaboration[N][S]            The sound settings window allows you
          /              \             to control your sound devices.
Open the control panel,   and click on the sound settings.
```

Planning with Rhetorical Structure Theory

RST (Mann and Thompson, 1988)

Planning with Rhetorical Structure Theory

RST (Mann and Thompson, 1988)

```
                                D
                                |
                        Background[N][S]
            
       Elaboration[N][S]              The sound settings window allows you
                                      to control your sound devices.

Open the control panel,   and click on the sound settings.
```

Planning with Rhetorical Structure Theory

RST (Mann and Thompson, 1988)

Planning with Rhetorical Structure Theory

Key idea: Grammar using RST relations (G_{RST})

Planning with Rhetorical Structure Theory

Key idea: Grammar using RST relations (G_{RST})

Assumption

Each record in the database input corresponds to a unique non-overlapping span in the collocated text, and can be therefore mapped to an EDU.

Grammar Extraction

desktop	start	start-target	window-target
Click start,	point to settings,	and then click control panel.	Double-click users and passwords.

contextMenu	action-contextMenu
On the advanced tab ,	click advanced.

Liang et al. (2009)

⇩

Grammar Extraction

desktop	start	start-target	window-target
Click start,	point to settings,	and then click control panel.	Double-click users and passwords.

contextMenu	action-contextMenu
On the advanced tab ,	click advanced.

Liang et al. (2009)

$\Downarrow$

[Click start,]^{desktop} [point to settings,]^{start} [and then click control panel.]$^{start-target}$
[Double-click users and passwords.]$^{window-target}$ [On the advanced tab,]^{contextMenu}
[click advanced.]$^{action-contextMenu}$

Grammar Extraction

D
|
Elaboration[N][S]

Elaboration[N][S]   Elaboration[N][S]

Elaboration[N][S]   Double-click users and passwords.   On the advanced tab,   click advanced.

Click start,   Joint[N][N]
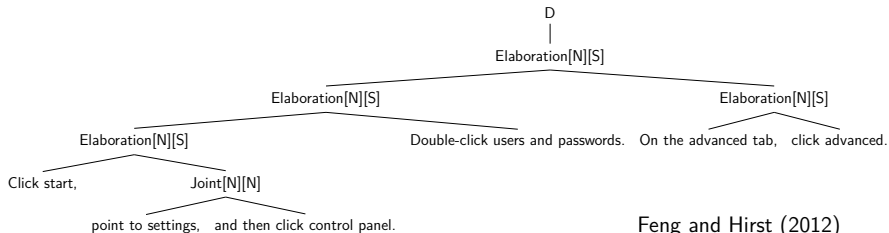
point to settings,   and then click control panel.   Feng and Hirst (2012)

Grammar Extraction

Feng and Hirst (2012)
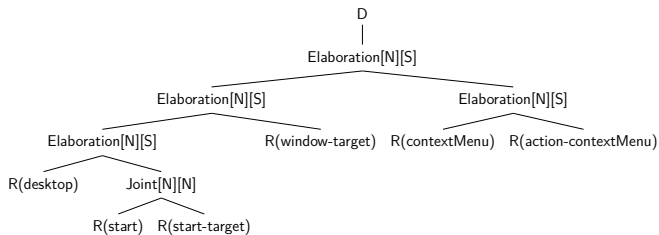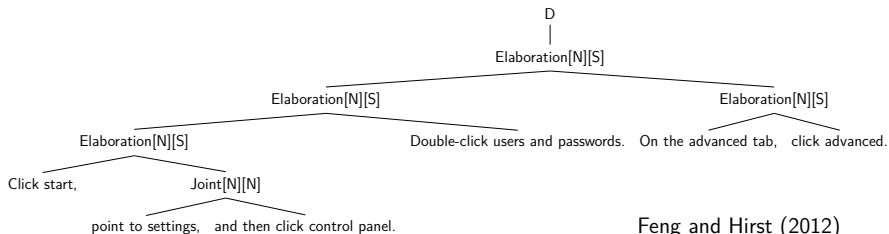
Extended Grammar

1. G_{RST}
2. $R(r_i.t) \rightarrow FS(r_j, start)$
3. $FS(r, r.f_i) \rightarrow F(r, r.f_j)FS(r, r.f_j) \mid F(r, r.f_j)$
4. $F(r, r.f) \rightarrow W(r, r.f)F(r, r.f) \mid W(r, r.f)$
5. $W(r, r.f) \rightarrow \alpha \mid g(f.v) \mid gen\_str(f.v, i)$

Experimental Setup

Data

- WEATHERGOV : weather reports [4 sents, 345 words]
  (Liang et al., 2009)
- WINHELP : troubleshooting guides [4.3 sents, 629 words]
  (Branavan et al., 2009)

Experimental Setup

Data

- WEATHERGOV : weather reports [4 sents, 345 words]
  (Liang et al., 2009)
- WINHELP : troubleshooting guides [4.3 sents, 629 words]
  (Branavan et al., 2009)

Evaluation

- Automatic evaluation: BLEU-4
- Human evaluation: Fluency, Semantic Correctness, Coherence

Experimental Setup

Data

- WEATHERGOV : weather reports [4 sents, 345 words]
  (Liang et al., 2009)
- WINHELP : troubleshooting guides [4.3 sents, 629 words]
  (Branavan et al., 2009)

Evaluation

- Automatic evaluation: BLEU-4
- Human evaluation: Fluency, Semantic Correctness, Coherence

System Comparison

- G_{RSE}, G_{RST}
- Konstas and Lapata (2012)
- Angeli et al. (2010)

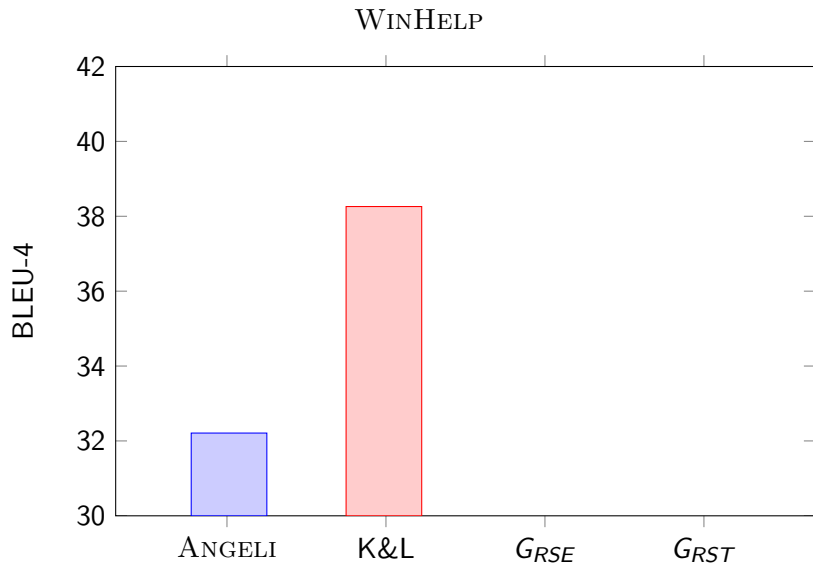# Results: Automatic Evaluation
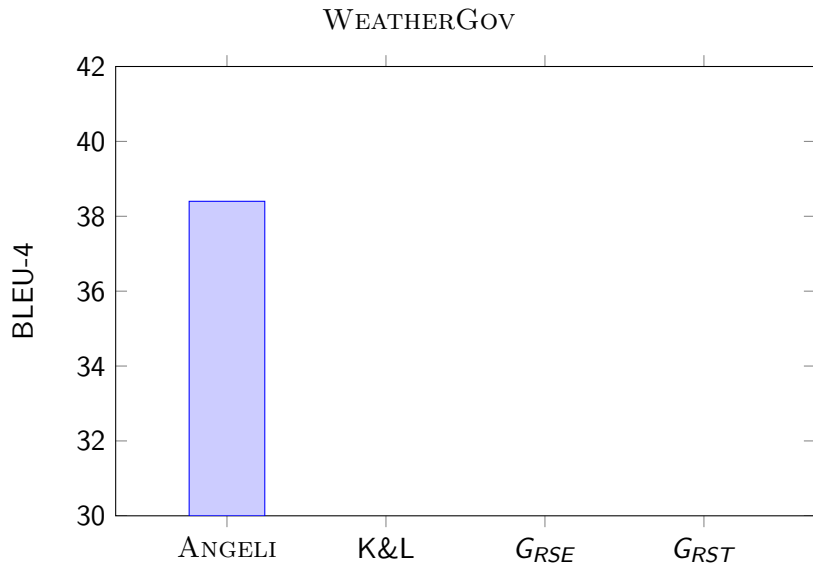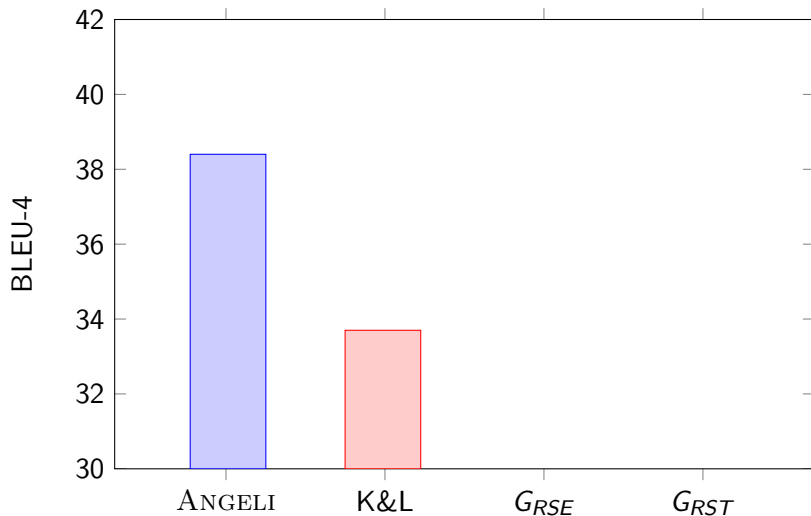
WEATHERGOV

Results: Automatic Evaluation

WeatherGov

BLEU-4 chart with bars for Angeli (~38.4) and K&L (~33.7); G_{RSE} and G_{RST} with no bars.

Results: Automatic Evaluation

WeatherGov

Results: Automatic Evaluation

WEATHERGOV

Results: Automatic Evaluation

WINHELP

BLEU-4 values for ANGELI, K&L, G_{RSE}, G_{RST} with y-axis from 30 to 42. ANGELI bar is at approximately 32.2.

Results: Automatic Evaluation

WINHELP

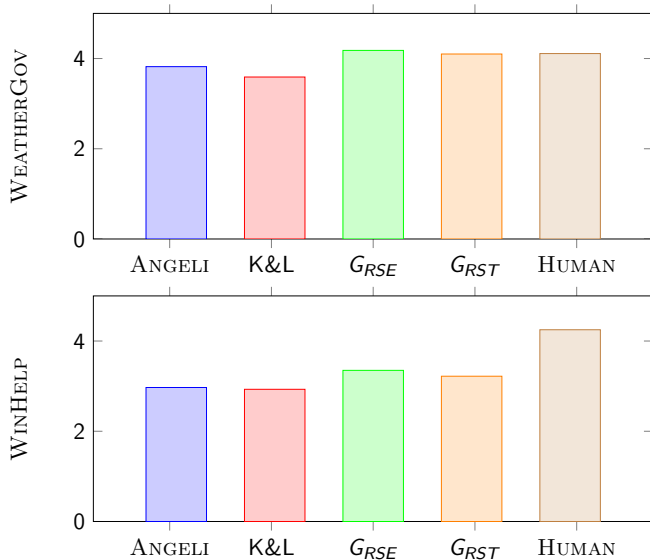# Results: Automatic Evaluation

WINHELP

Results: Automatic Evaluation

WINHELP

BLEU-4 chart with values for ANGELI, K&L, G_{RSE}, G_{RST}
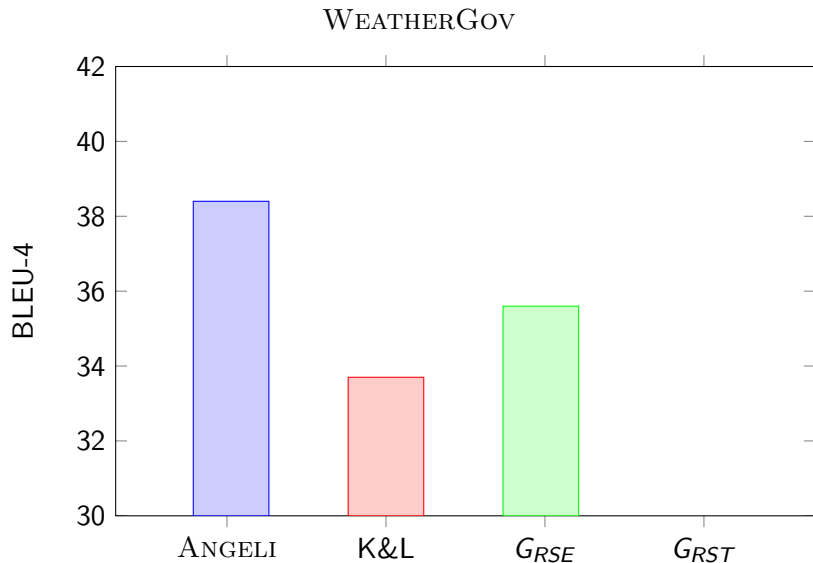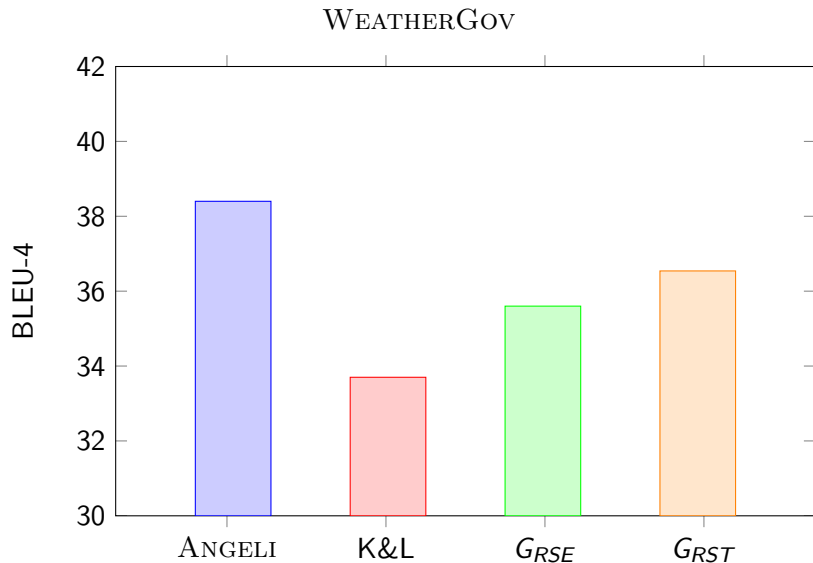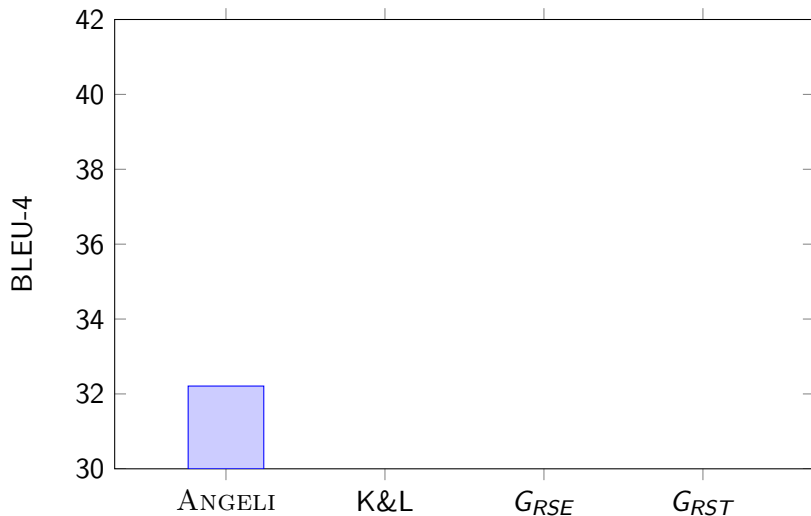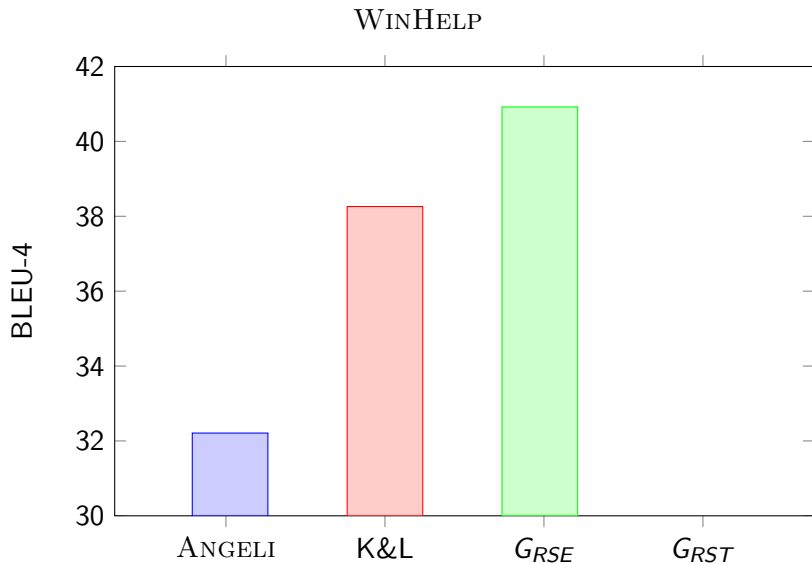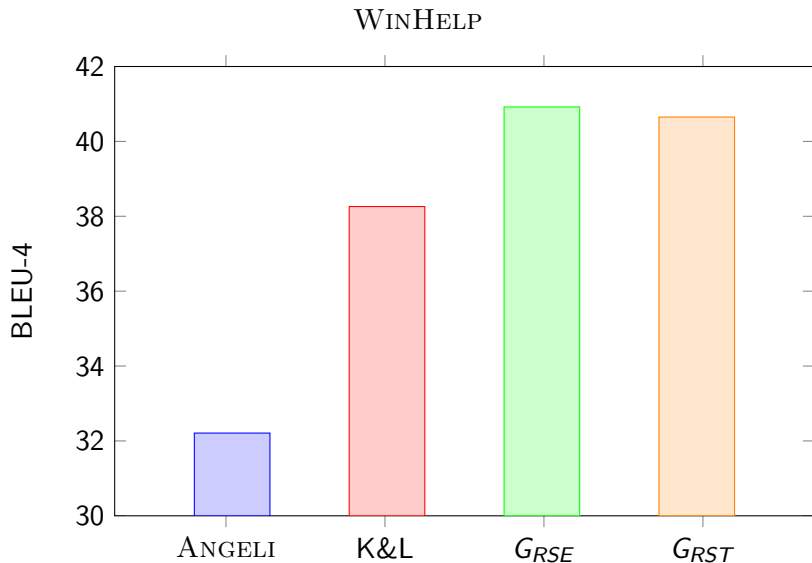
Results: Human Evaluation (Coherence)

Output

G_{RSE}	Click start, point to settings, and then click control panel. Double-click network and dial-up connections. Right-click local area connection, and then click properties. Click install, and then click add. Click network monitor driver, and then click ok.
K&L	Click start, point to settings, and then click control panel. Double-click network and dial-up connections. Double-click network and dial-up connections. Right-click local area connection, and then click ok.
HUMAN	Click start, point to settings, click control panel, and then double-click network and dial-up connections. Right-click local area connection, and then click properties. Click install, click protocol, and then click add. Click network monitor driver, and then click ok.

Conclusions

- End-to-end generation system that incorporates document planning
- Grammar-based approach allows for document planning naturally: all we need is a discourse grammar
- Provide two solutions for document plans:
    - Linguistically naive record sequence grammar (G_{RSE})
    - RST-inspired grammar (G_{RST})

- Future work: more challenging domains (financial, biographies)

Thank you

Questions ?

Decoding

$$\hat{g} = f\left(\arg\max_{g,h} p(g) \cdot p(g, h \mid \mathbf{d})\right)$$

- Bottom-up Viterbi search
- Keep k-best derivations at each node, cube pruning (Chiang, 2007)
- $p(g)$ rescores derivations using an n-gram language model