

Detection of Conditionals in Spoken Utterances

Sebastian Weigelt, Tobias Hey, and Vanessa Steurer

KIT – Department of Informatics – Institute for Program Structures and Data Organization (IPD).



Intelligent Assistants seem to be
quite smart today!

How far away is the sun?

When is my next meeting?

Is my daughter at home?

Set a timer for five minutes

Call my brother at work

How many dollars is 45 euro

Remind me to call mom

Google the war of 1812

Who is near me?

Text Brian I'm on my way

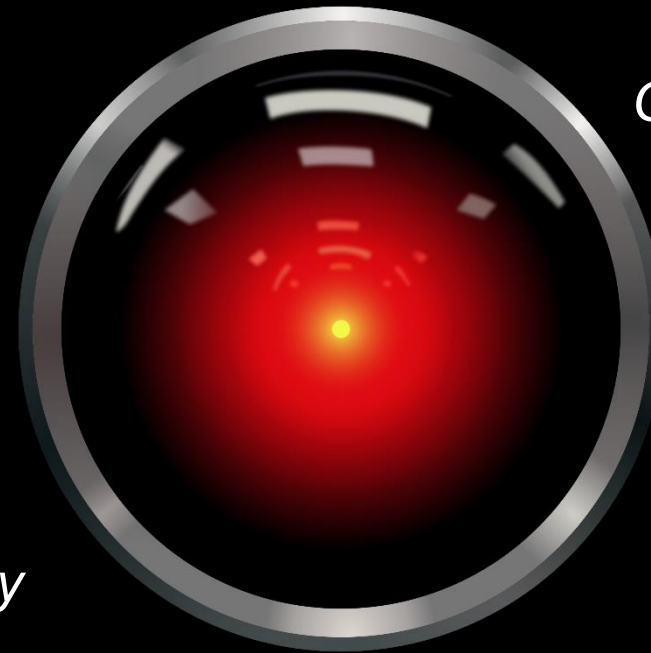
Give me directions home

Find Disney movies

Play iTunes Radio

Should I bring an umbrella?

What's trending on twitter?



But do they really understand?

“book a business class flight to LA”

“if no business class
tickets are available book economy”

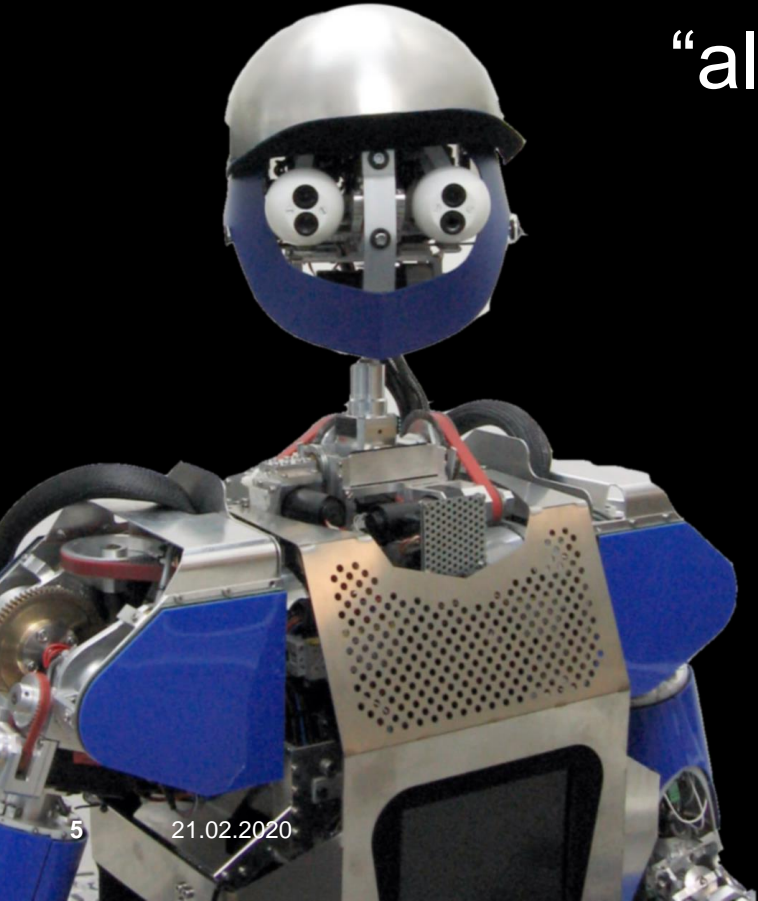
“also order a Wi-Fi
pass for me”



“when I leave do the laundry”

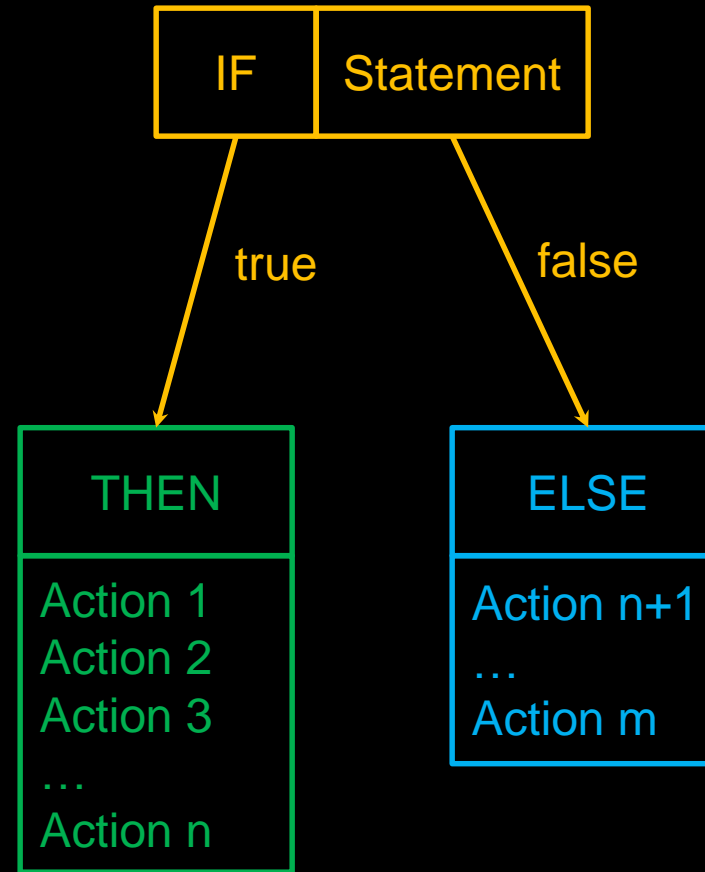
“iron it”

“and fold it”



“book a business class flight to LA

if no business class tickets...”



```

If (ticket.available()){
    ticket.book();
    ...
} else {
    ...
}
  
```



Conditionals

- two-clause structure
 - conditional clause (introduced by keyword)
 - dependent clause
 - then-clause (mandatory, **may be** introduced by keyword)
 - else-clause (optional, introduced by keyword)
 - syntactic structure: if-then(-else) or then-if(-else)
- types
 - premise-conditionals
 - event-conditionals
- reference frame
 - Which phrases depend on which conditional?

*“when I leave
do the laundry...”*

*“iron it
fold it
go to
...” the cupboard*

Reference Frame

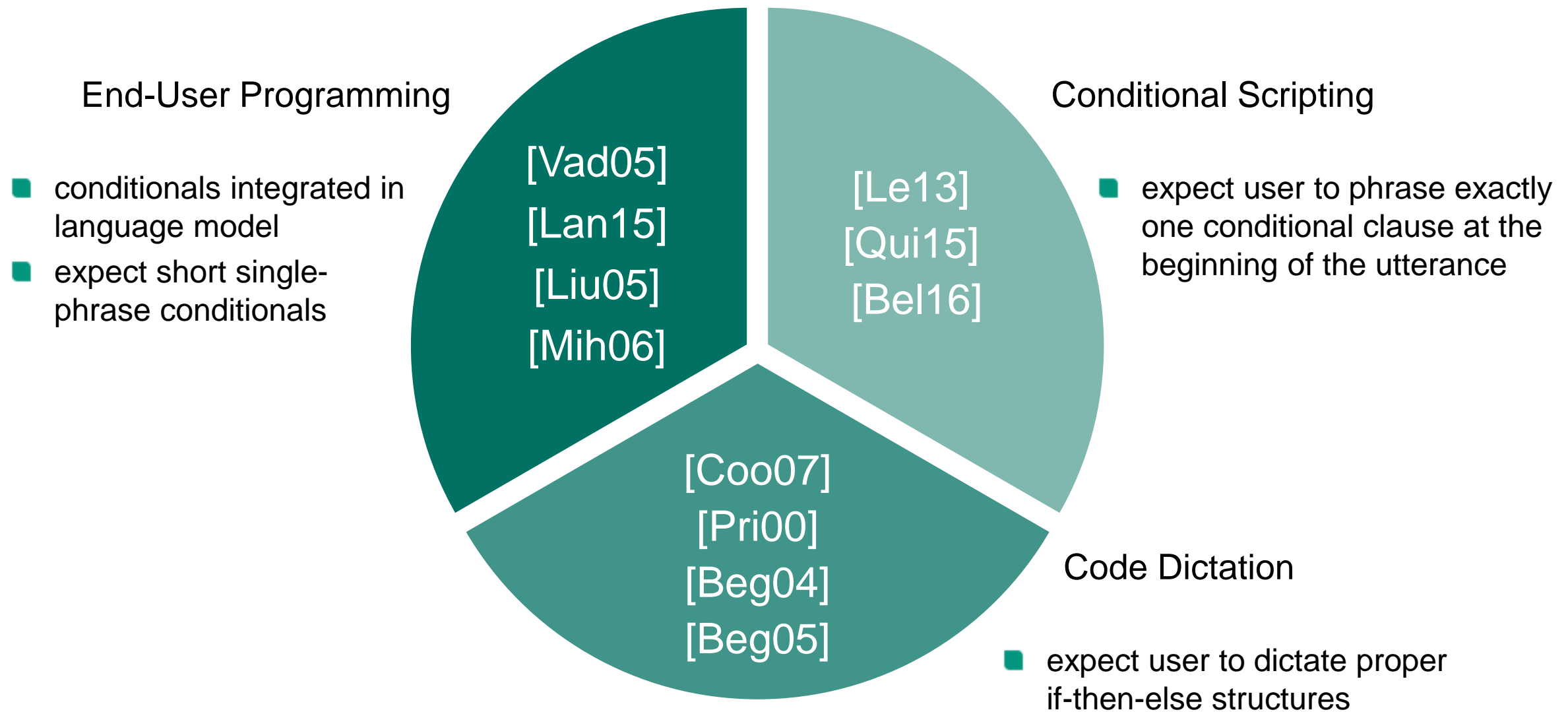
- characterizes the boundaries of the conditional structure
 - the **extent** of the **conditional clause**
 - the **position** (before or after the conditional clause) and **extent** of the **then-clause**
 - the **existence** and **extent** of the **else-clause**

*“when I leave
do the laundry
iron it
fold it”*



```
ReferenceFrame(Cond0)
COND:
  I leave
THEN
  do the laundry
  iron it
  fold it
```

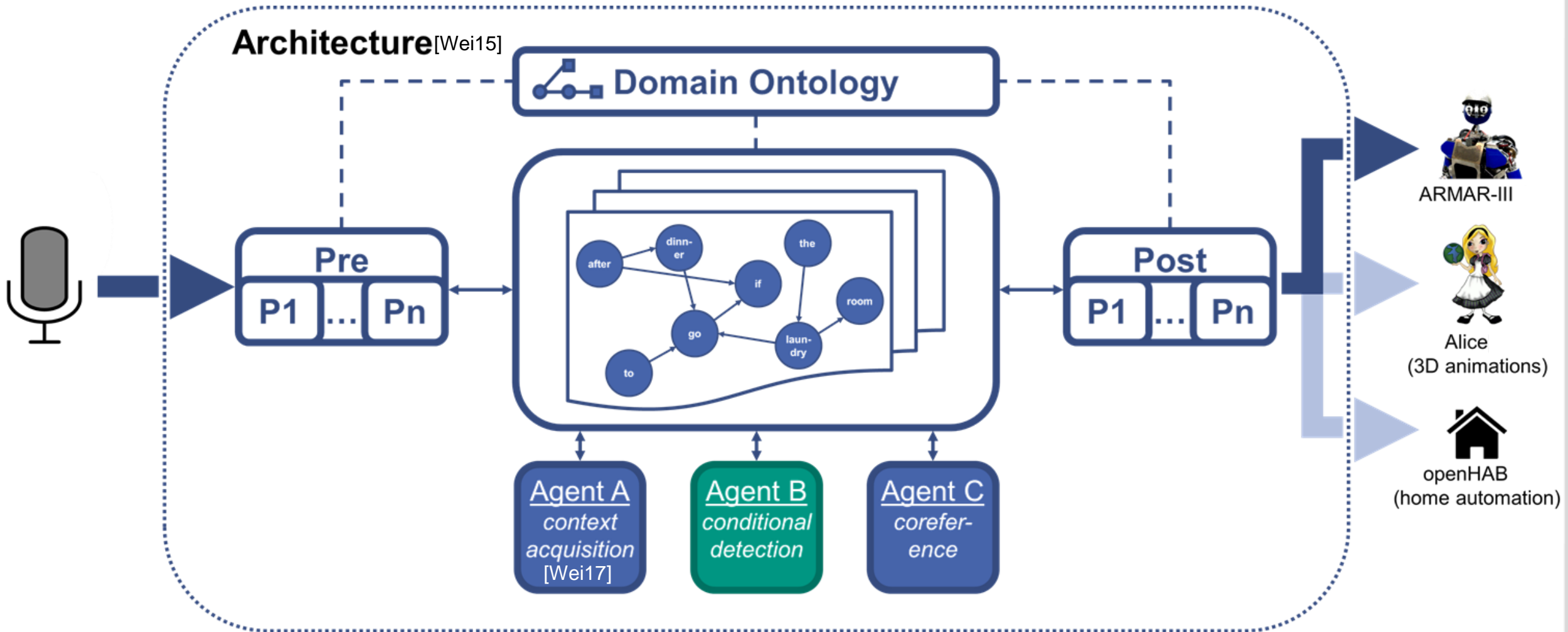

Related Work



Detection of Conditionals

- Objective
 - semantic representation of conditionals
- Approach
 - use syntax to detect basic conditional structures and
 - context to determine reference frame
- Setting
 - End-User Programming with spoken language

PARSE



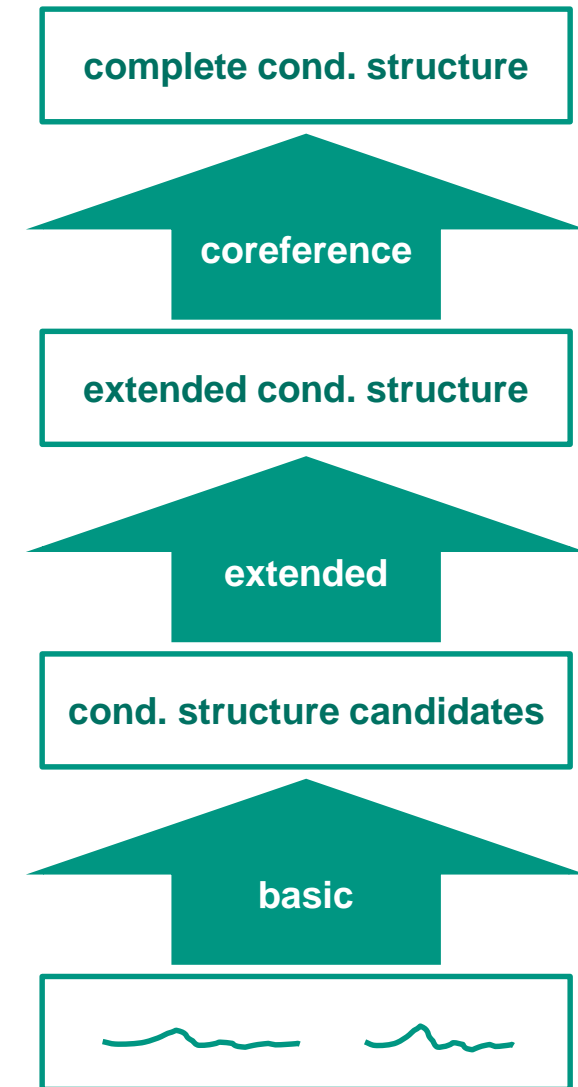
Detection of Conditionals – Approach

■ Challenge



■ Approach

- keywords & robust grammars
- extended grammars
- coreference information analysis

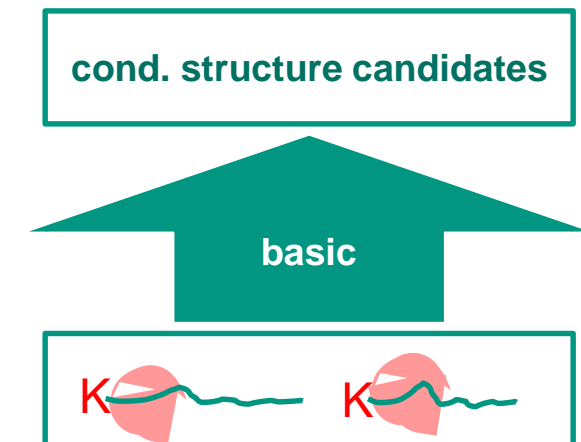


Detection of Conditionals – Keywords and robust grammars

- Use keywords to spot...
 - conditional and else-clauses
 - then-clauses (in case keywords exist)

- Use robust grammars to...
 - build “one phrase” candidates
 - verify or discard keywords

Clause Type	Keywords/-phrases
cond. clause	if, when, suppose(d) that, supposing that, whenever, in case, in the case that, ...
then-clause	then, please, if so, you can, you have to, could you, ...
else-clause	else, if not, otherwise, elseways, alternatively, ...



Detection of Conditionals – Extended Grammars

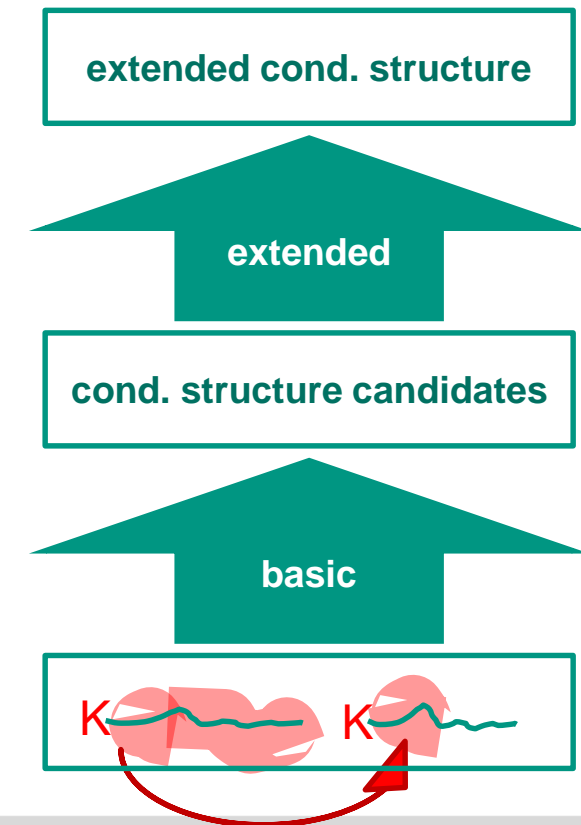
- use extended grammars to...
 - verify candidate clauses
 - generate coherent structures: if-then-else
 - expand structures

- Fallback: use results produced by robust grammars

snippet of the advanced grammar for conditional clauses (if-clauses):

```

conditional → if-clause then-clause
conditional → if-clause then-clause else-clause
if-clause   → if-keyword NBS
NBS         → NPB VPB | NPB VPB Conj NPB VPB
NPB         → NP CC NPB | NP PP NPB | NP
VPB        → VP CC VPB | VP PP NPB | VP PP VPB
            | ADVP VPB | VP VMD | VP
CC          → Conj | Neg
VMD         → ADJP | ADVP | PRT
  
```



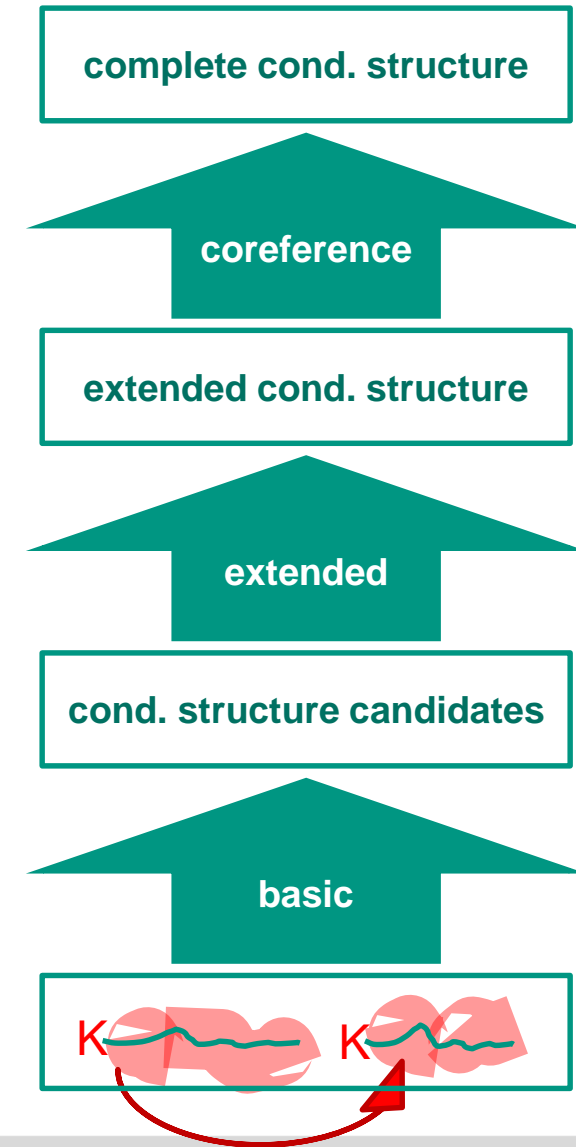
Detection of Conditionals – Coreference Information Analysis

- use coreference information
 - provided by an PARSE agent
 - to expand structures even further

- Idea: **entities mentioned multiple times in a row belong to the same type of clause structure**

- Approach
 1. find all coreference chains
 2. add all phrases that are framed by the chain to the clause structure (defined by the first phrase)
 3. if a keyword is present: split chain

“If the laundry isn’t dry yet you have to wait
for it otherwise iron it and fold it”



Application

“after dinner

go to the laundry room

if the laundry is done

iron the shirts

and fold them

otherwise

come back to the kitchen”

```
robot.goTo(laundryRoom);
```

```
robot.iron(shirts);
```

```
robot.fold(them);
```

```
robot.goTo(kitchen);
```


Application

“after dinner

go to the laundry room

if the laundry is done

iron the shirts

and fold them

↑ coref

otherwise

come back to the kitchen”

```
robot.goTo(laundryRoom);
```

```
robot.iron(shirts);
```

```
robot.fold(them);
```

```
robot.goTo(kitchen);
```

Application

“after dinner

go to the laundry room

if the laundry is done

iron the shirts

and fold them

↑ coref

otherwise

come back to the kitchen”

```
robot.goTo(laundryRoom);
```

```
robot.iron(shirts);
```

```
robot.fold(shirts);
```

```
robot.goTo(kitchen);
```

Application

“after dinner

go to the laundry room

if the laundry is done

iron the shirts

and fold them

↑coref

otherwise

come back to the kitchen”

```
robot.goTo(laundryRoom);
```

```
if laundry.state = done then
```

```
    robot.iron(shirts);
```

```
    robot.fold(shirts);
```

```
else
```

```
    robot.goTo(kitchen);
```

```
end if
```

Evaluation – Setting



- 19 subjects
- 2 tasks
- Long and complex utterances

robo go to the table if there are any dirty dishes grab the dirty dishes and go to the dishwasher open the dishwasher and put the dirty dishes into the dishwasher close the dishwasher and return to the table if there are any clean dishes grab the clean dishes and go to the cupboard open the cupboard and put the clean dishes into the cupboard

	Scenario 1	Scenario 2	Total
Recordings	19	17	36
Words	556	538	1094
Conditionals	28	19	47
Recordings w/o conditionals	4	2	6

Evaluation – Results

	Grammar		
	Precision	Recall	F ₁
m. transcription	0.930	0.803	0.862
Google Speech ¹	0.817	0.670	0.736
IBM Watson ASR ²	0.862	0.665	0.751

¹<https://cloud.google.com/speech/>, 2017-10-04

²<https://www.ibm.com/watson/developercloud/speech-to-text.html>, 2017-10-04

Evaluation – Results

	Grammar			Coreference		
	Precision	Recall	F ₁	Precision	Recall	F ₁
m. transcription	0.930	0.803	0.862	0.934	0.864	0.898
Google Speech ¹	0.817	0.670	0.736	0.824	0.704	0.760
IBM Watson ASR ²	0.862	0.665	0.751	0.869	0.712	0.783

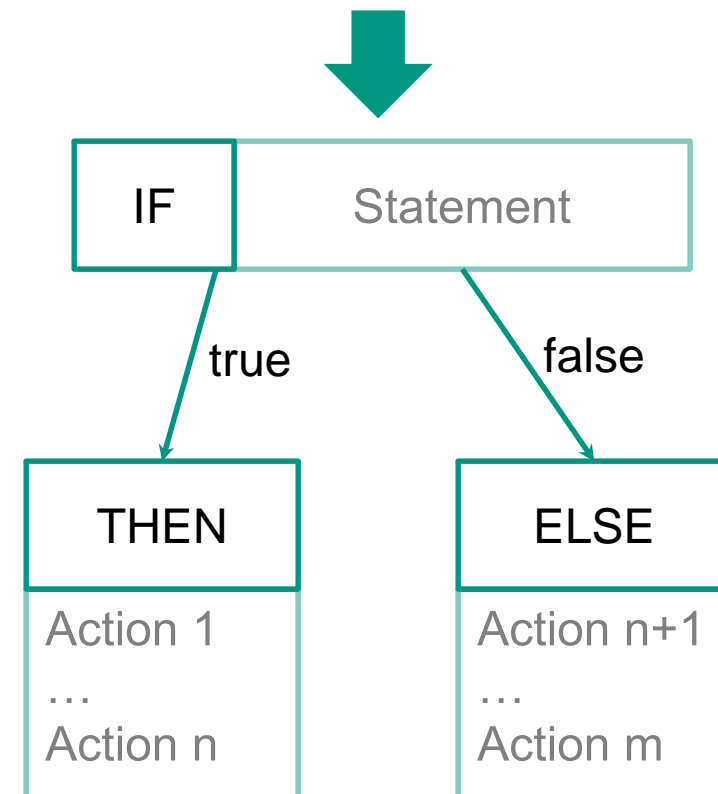
¹<https://cloud.google.com/speech/>, 2017-10-04

²<https://www.ibm.com/watson/developercloud/speech-to-text.html>, 2017-10-04

Conclusion

- Objective: detection of conditionals
 - composed of conditional clauses (*if*) and
 - dependent clauses (*then* and *else*)
 - and create a semantic representation

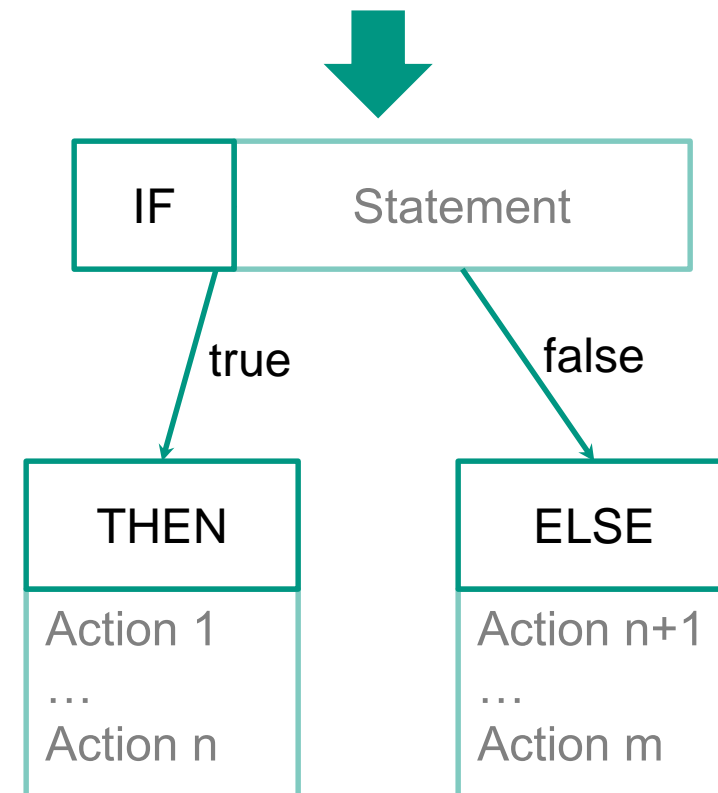
*“if the laundry is done iron the shirts
and fold them otherwise come back to
the kitchen”*



Conclusion

- Objective: detection of conditionals
 - composed of conditional clauses (*if*) and
 - dependent clauses (*then* and *else*)
 - and create a semantic representation
- Approach: determine reference frame
 - with grammars and
 - coreference information

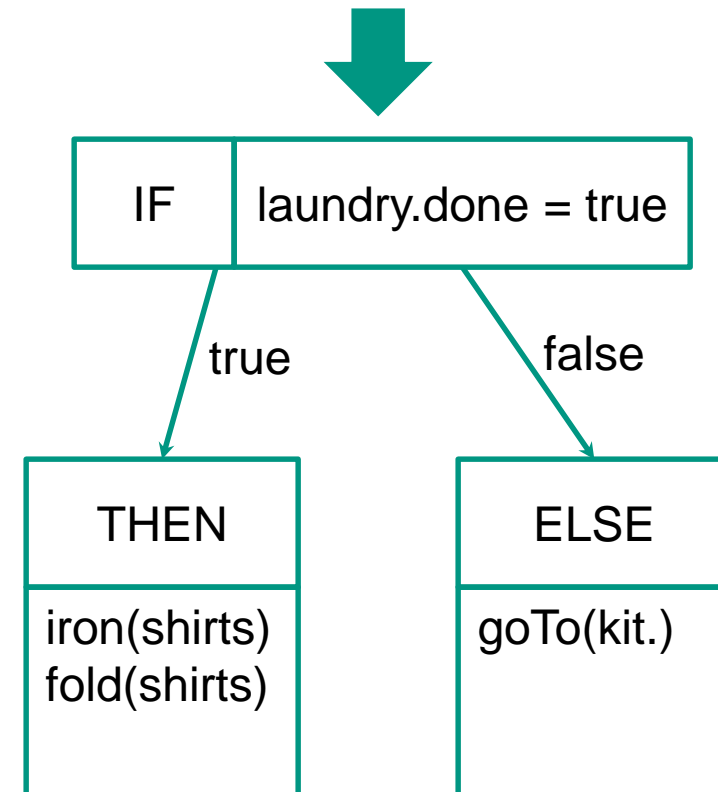
*“if the laundry is done iron the shirts
and fold them otherwise come back to
the kitchen”*



Conclusion

- Objective: detection of conditionals
 - composed of conditional clauses (*if*) and dependent clauses (*then* and *else*)
 - and create a semantic representation
- Approach: determine reference frame
 - with grammars and coreference information
- Evaluation: user study
 - manual transcripts: $F_1 = 90\%$
 - automatic speech recognition: $F_1 = 78\%$
- Future Work
 - Extension (other control structures: loops, parallelism, etc.)
 - Statistical methods (machine learning)

“if the laundry is done iron the shirts and fold them otherwise come back to the kitchen”



References (1)

- [Coo07] W. R. Cook, “Applescript,” in Proceedings of the Third ACM SIGPLAN Conference on History of Programming Languages, ser. HOPL III. New York, NY, USA: ACM, 2007.
- [Pri00] D. Price, E. Riloff, J. Zachary, and B. Harvey, “NaturalJava: A Natural Language Interface for Programming in Java,” in Proceedings of the 5th International Conference on Intelligent User Interfaces, ser. IUI '00. New Orleans, Louisiana, USA: ACM, 2000
- [Beg04] A. Begel, “Spoken Language Support for Software Development,” in 2004 IEEE Symposium on Visual Languages and Human Centric Computing, Sep. 2004
- [Beg05] A. Begel and S. Graham, “Spoken programs,” in 2005 IEEE Symposium on Visual Languages and Human-Centric Computing, Sep. 2005
- [Le13] V. Le, S. Gulwani, and Z. Su, “Smartsynth: Synthesizing Smartphone Automation Scripts from Natural Language,” in MobSys'13, vol. 2, 2013
- [Qui15] C. Quirk, R. Mooney, and M. Galley, “Language to code: Learning semantic parsers for if-this-then-that recipes,” in Proceedings of the 53th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, 2015
- [Bel16] I. Beltagy and C. Quirk, “Improved semantic parsers for if-then statements,” in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, 2016

References (2)

- [Vad05] D. Vadas and J. R. Curran, “Programming With Unrestricted Natural Language,” in Proceedings of the Australasian Language Technology Workshop, Sydney, Australia, Dec. 2005
- [Lan15] M. Landhäußer and R. Hug, “Text Understanding for Programming “in Natural Language: Control Structures,” in Proceedings of the 4th International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering, May 2015
- [Liu05] H. Liu and H. Lieberman, “Metafor: Visualizing Stories As Code,” in Proceedings of the 10th International Conference on Intelligent User Interfaces, ser. IUI '05. New York, NY, USA: ACM, 2005
- [Mih06] R. Mihalcea, H. Liu, and H. Lieberman, “NLP (Natural Language Processing) for NLP (Natural Language Programming),” in Proceedings of the 7th International Conference on Computational Linguistics and Intelligent Text Processing, ser. CICLing'06, 2006
- [Wei15] S. Weigelt and W. F. Tichy, “Poster: ProNat: An Agent-Based System Design for Programming in Spoken Natural Language,” in 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering (ICSE), vol. 2, May 2015
- [Wei17] S. Weigelt, T. Hey, and W. F. Tichy, “Context Model Acquisition from Spoken Utterances,” in The 29th International Conference on Software Engineering & Knowledge Engineering, Pittsburgh, PA, Jul. 2017