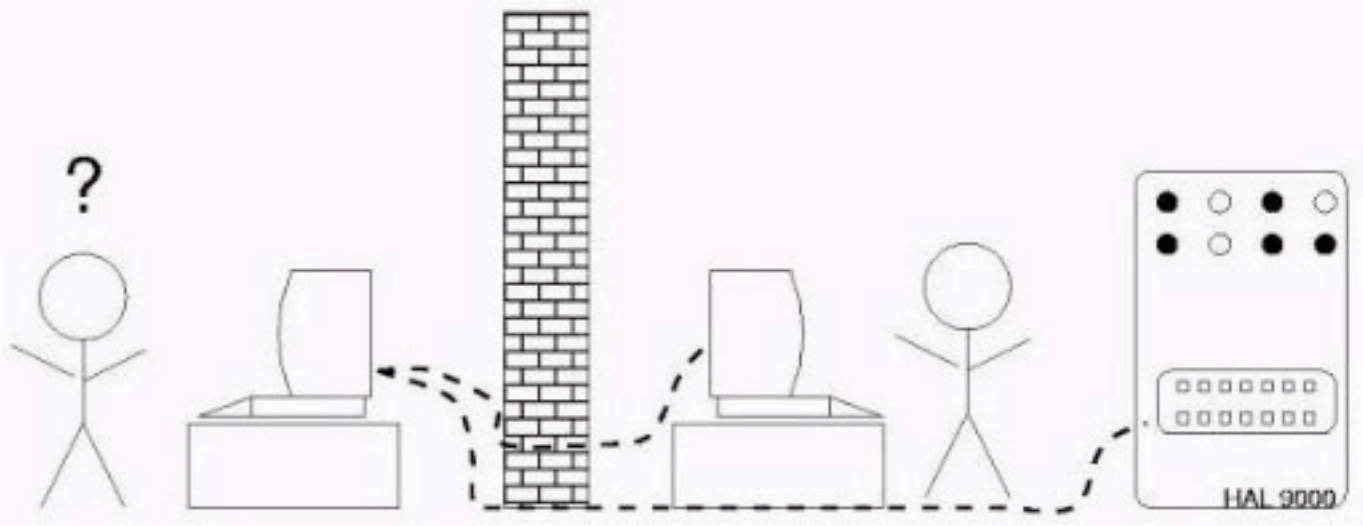
Elements of Data Science and Artificial Intelligence

WS 2019/20

Vera Demberg

Turing Test (you've seen this slide before)

Take 1: The Turing Test



- Suggested (in various forms) by Alan Turing as a measurable definition of AI.

Natural Language

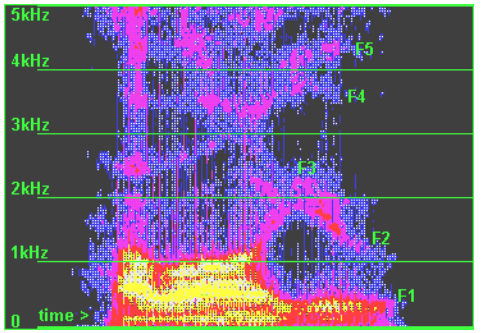
- Natural Language is the most complex human ability
- Central to many AI tasks and applications
- Goals of the next six lectures
  - Understanding challenges related to language processing
  - Insight into how natural language works
  - Modelling and representing natural language
  - Machine learning methods for natural language processing
  - Example application: dialog systems

Language assistants

Google Home Amazon Echo

What is language processing?

Sound signal



Speech recognition

Laura schläft

Word sequence

Language understanding

Information

What is language processing?

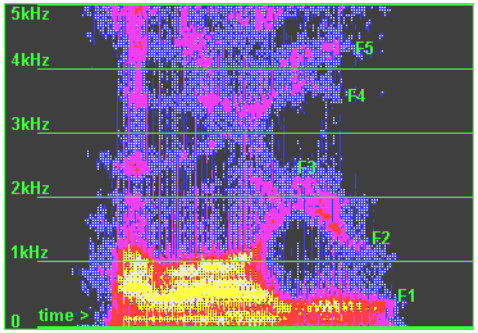
Information

Language generation

Word sequence

Laura schläft

Speech synthesis



Sound signal

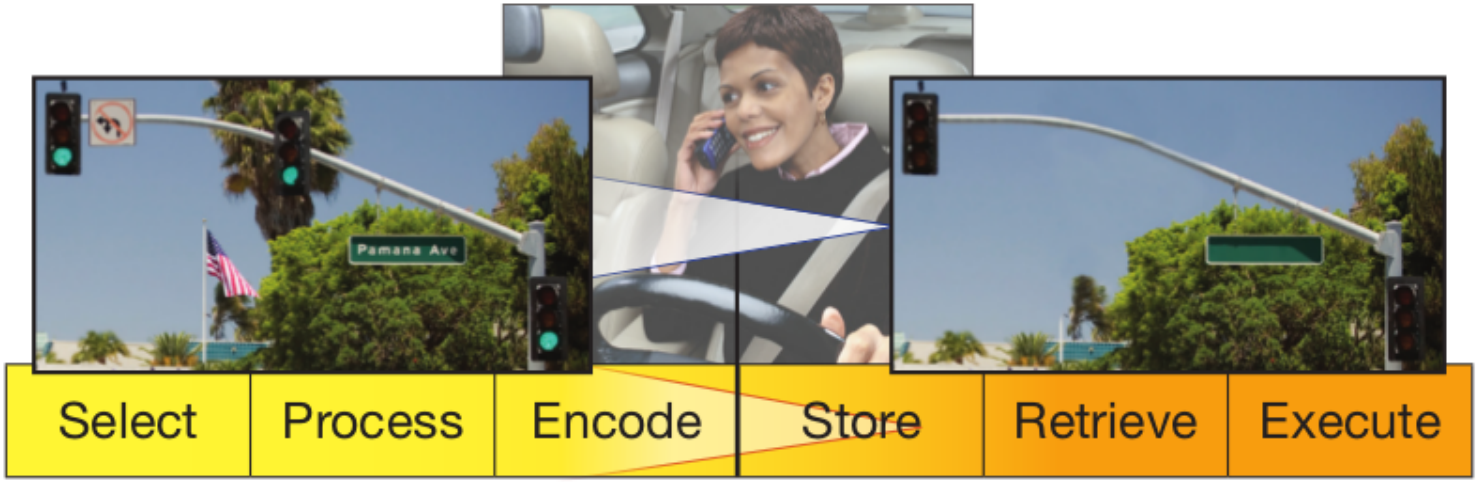
The role of context in dialogue

Google Home Amazon Echo

In-car dialog

New challenges

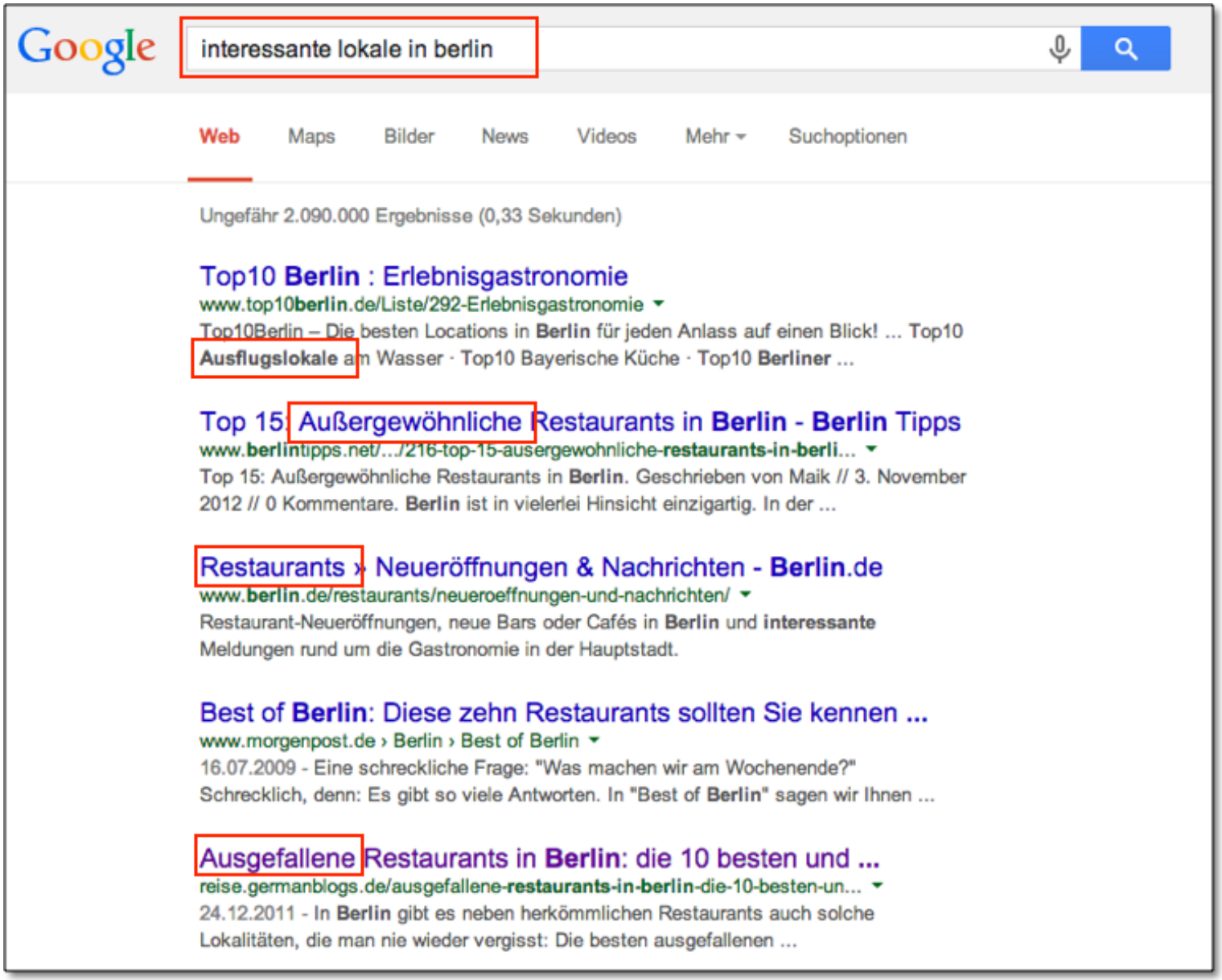
New developments also bring new challenges.



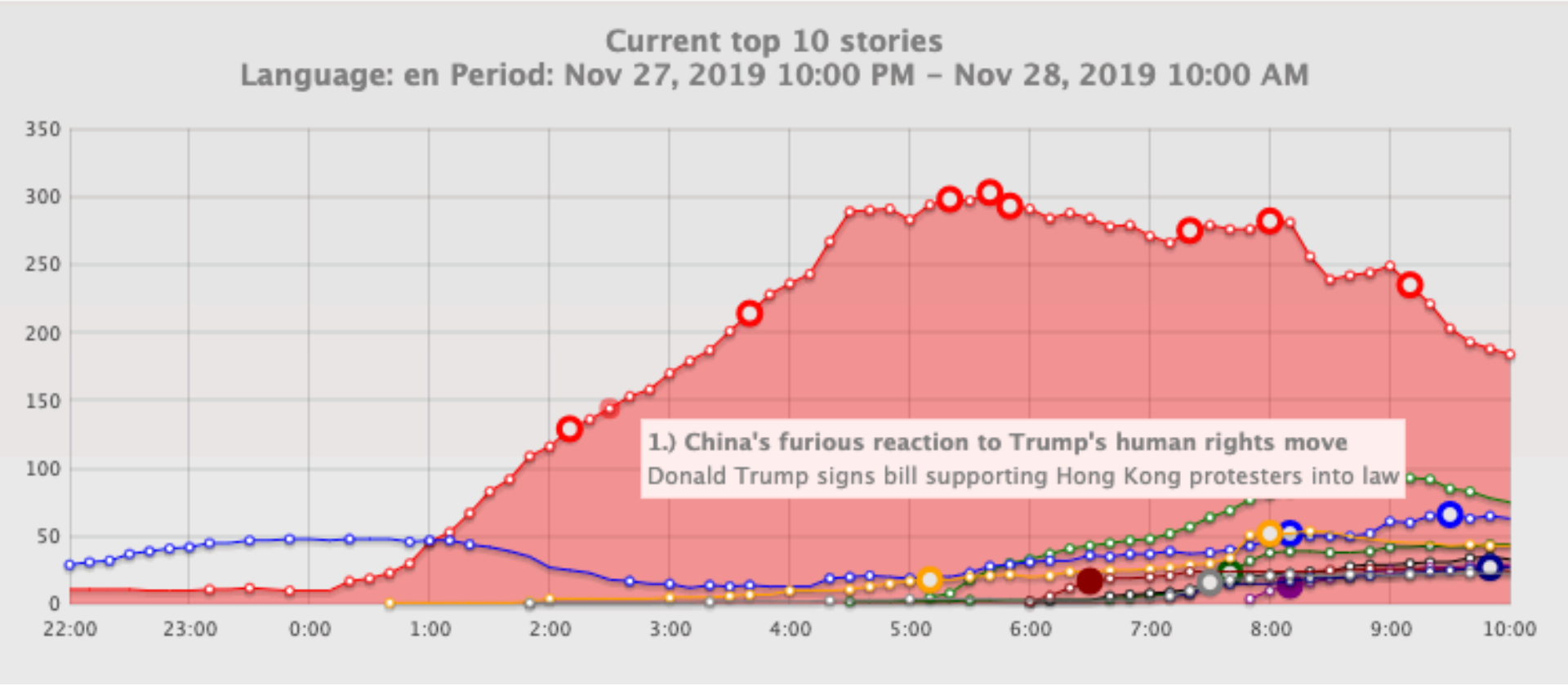
How can we design language interaction such that usage is safe?

Driving and dialogue lab

Other examples for NLP applications

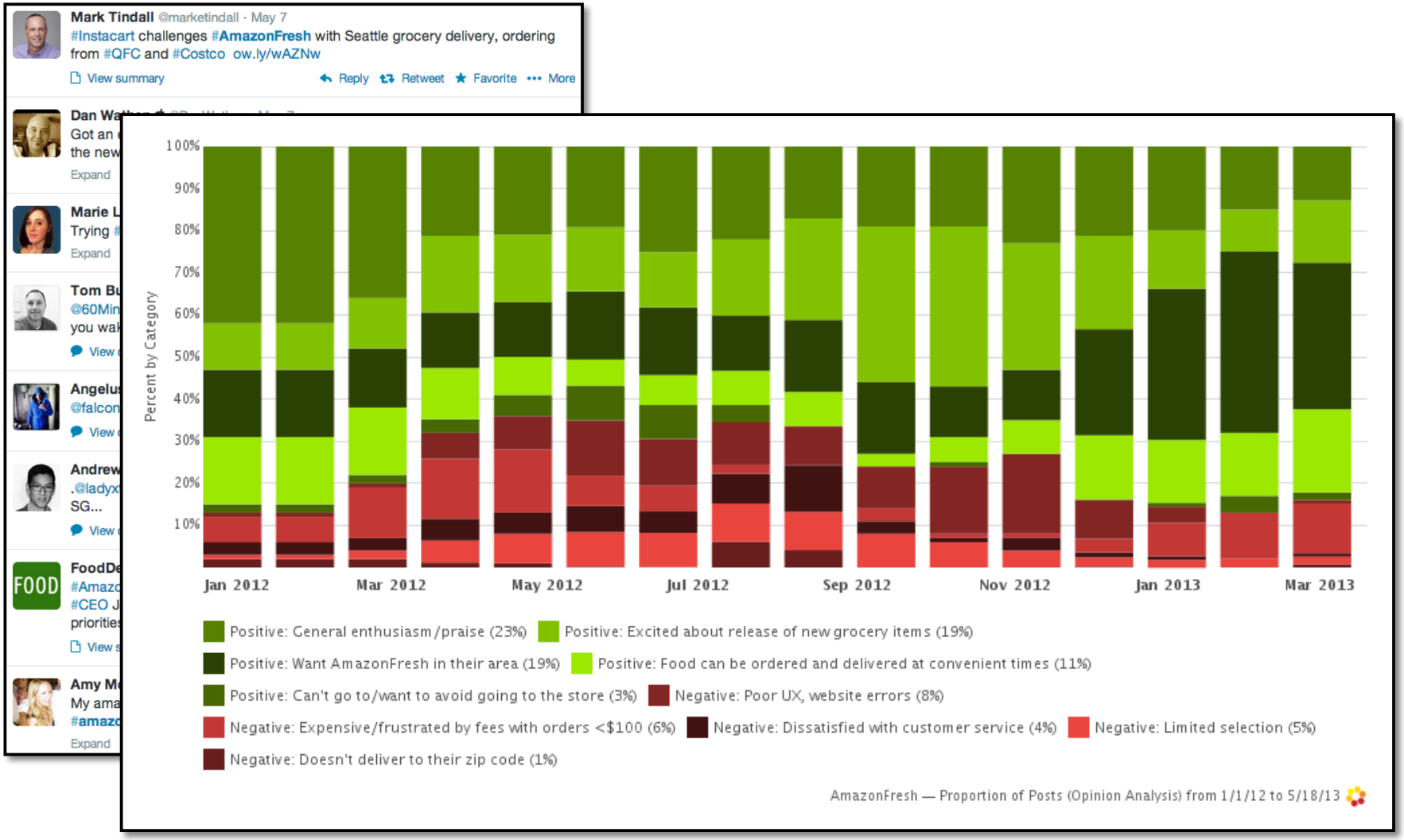


News summarization



European Media Monitor, http://emm.newsbrief.eu

CL im (Business-) Alltag



Crimson Hexagon, http://www.crimsonhexagon.com/

Multilingual applications

For some applications, it is necessary to work with more than just one language

- Translation
- Information retrieval across languages
- Language tutoring systems

Google Translate

Gewalt wegen Kupferhütte

Tödliche Umweltproteste in Indien

Deutsch

Wir demonstrieren weiter, soll die Polizei uns doch töten."

Englisch

We continue to demonstrate that the police should kill us. "

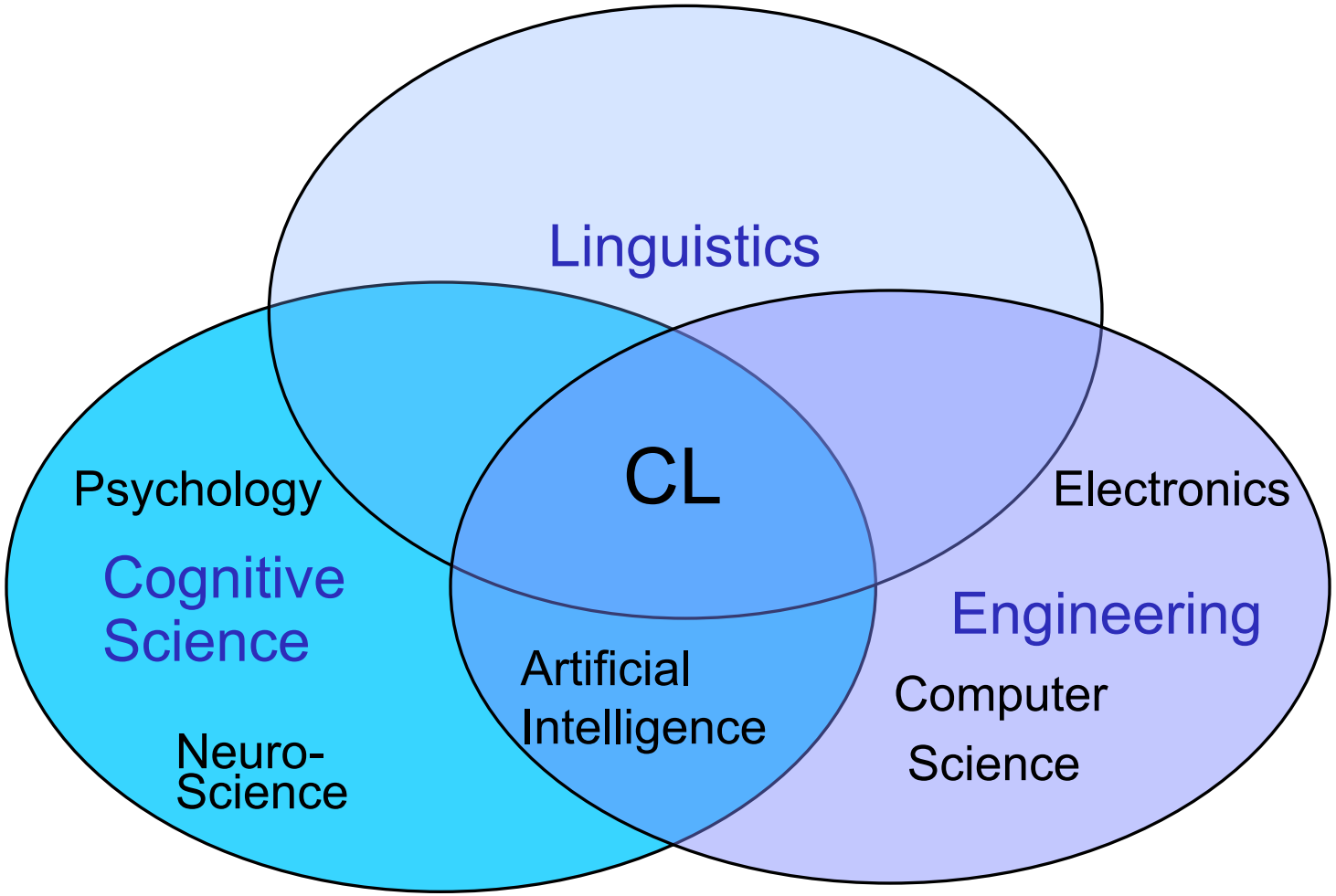
Tasks in natural language processing

- Modelling of complex language processes:
  - language understanding
  - language production
  - language acquisition

- Development of formalisms and tools for language representation, language processing at various linguistic levels:
  - Phonetics and phonology
  - Morphology and syntax
  - Semantics
  - Pragmatics, Text and Dialog

- Development of tools for language processing.

Computational Linguistics



What is language processing?
Contents

12.12.	Introduction to language processing and dialogue
16.12.	Syntax and semantics of natural language
19.12.	Language models
6.1.	(Neural) natural language generation
9.1.	Dialogsystems
13.1.	Pragmatics (situated language processing)

In this course, we will place a major focus on dialogue systems, as one example for an NLP application.

History of natural language processing

- In the 1950s, the US promoted the development of machine translation (for the language pair Russian-English) for political reasons.
  The results were extremely bad and the funding was discontinued.

- In the late 60s and early 70s, a second generation of natural language processing systems started to develop: dialog systems.

  The two most famous „historical“ dialog systems are:
  - ELIZA, 1966, by Joseph Weizenbaum
  - SHRDLU, 1970, by T. Winograd

ELIZA: Beispieldialog

(1) U: I have been very depressed lately

S: *You say you have been very depressed lately.*

(2) U: Yes, my mother beats me and my father calls me names

S: *You`re being somewhat short with me*

(3) U: What about my mother beating me?

S: *Tell me more about your family*

(4) U: My father calls me nasty names

S: *Who else in your family calls you nasty names*

(5) U: I think the dog would if he could

S: *Do you think its likely that he could*

(6) U: No, you do

S: *You don`t seen very talkative today*

ELIZA: Beispieldialog [2]

(7) U: I guess what is really bothering me is that I don`t have any money

S: *Can you elaborate on that*

(8) U: The problem is that I know psychiatrists are expensive and I may not be able to pay you

S: *Please continue, this may be interesting*

(9) U: I thought you might think so

S: *Perhaps in your fantasy we thought each other*

Was steckt hinter ELIZA?

TEMPLATE: `I _X1_`
RESPONSE: `You say you _X1_`

TEMPLATE: `_X1_ my _X2_(category family) _X3_`
RESPONSE: `Who else in your family _X3_ ?`

TEMPLATE: `_X1_ if _X2_`
RESPONSE: `Do you think its likely that _X2_ ?`

TEMPLATE: `_X1_`
RESPONSE: `You‘re being somewhat short with me.`

Regular expressions

Regular expressions are useful for matching patterns in texts.

For example:

- Find all words that contain a hyphen “-”
- Find all verbs that end in “ing”
- Identify all telephone numbers that are contained in a text collection
- Find and remove all html tags from a text

Regular Expressions

For simple tasks, one can just search (e.g. for “-”).

But more complex tasks such as finding all participle forms with ge...t as in “gearbeitet” or “gefreut” in German require more complex patterns.

It can therefore be helpful to be able to generalize across character classes.

Special symbols in regular expressions

[

\

^

$

.

|

?

*

+

( )

Sonderzeichen bei regulären Ausdrücken

[ ] in eckigen Klammern können wir Zeichenmengen definieren. Der Bindestrich gibt dabei einen Bereich an.

z.B.: [a-z] bezeichnet einen beliebigen Kleinbuchstaben zwischen a und z.

Achtung: Umlaute und ß sind nicht drin!

[a-zäöüß]

[a-e] bezeichnet entsprechend einen beliebigen Kleinbuchstaben zwischen a und e

[0-9] eine beliebige Ziffer

[aeiou] ein Vokal

[Gg]r[ae]y matcht alle Vorkommen von “grau” auf Englisch

Sonderzeichen bei regulären Ausdrücken

\ Der Backslash wird benutzt, wenn man eines der Sonderzeichen im regulären Text finden möchte (also es nicht als Sonderzeichen benutzen möchte).

z.B. 1\+1=2

Abkürzungen:

\d	digit = Ziffer, das gleiche wie [0-9]
\w	Wort = [A-Za-z0-9_]
\s	space = [\t\r\n]
\D	keine Ziffer = [^\d]
\W	kein Wort = [^\w]
\S	kein Leerzeichen o.ä.: [^\s]

Sonderzeichen bei regulären Ausdrücken

Abkürzungen:

\d digit = Ziffer, das gleiche wie [0-9]

\w word = [A-Za-z0-9_]

\s space = [\t\r\n]

\D keine Ziffer = [^\d]

\W kein Wort = [^\w]

\S kein Leerzeichen o.ä.: [^\s]

Sonderzeichen bei regulären Ausdrücken

^ hat zwei Bedeutungen:

- am Anfang von eckigen Klammern bedeutet es “nicht”.
  z.B.: q[^u] (finde alle q die danach kein u haben).
  Iraqi citizens have quite a hard time obtaining a visa.

- Ansonsten bedeutet es “Zeilenanfang”.
  z.B.: ^H
  Heute haben wir großen Hunger.

Sonderzeichen bei regulären Ausdrücken

$ Zeilenende

\b Wortanfang / Wortende
\B nicht am Wortanfang / Wortende

\bist\b

Der Mann ist an einer Reistonne.

\Bist\B matcht *ist* in *Reistonne*.

Sonderzeichen bei regulären Ausdrücken

. “irgendetwas”

Der Punkt kann ein beliebiges Zeichen matchen.

e.g. “ge....t” matcht *gezeigt*, *gefragt*, *gemacht etc.*

“gehungert” wird allerdings nicht gematcht, da wir genau 4 Punkte angegeben hatten, d.h. wir können nur genau 4 Zeichen matchen.

Achtung, wir könnten auch “gebt Atome” matchen!

Sonderzeichen bei regulären Ausdrücken

Wie oft soll etwas vorkommen?

? Ein mal oder kein mal

* Egal wie oft (oder gar nicht)

+ Egal wie oft, aber mindestens ein mal.

{} Anzahl kann in den geschweiften Klammern angegeben werden.

“Schifff?ahrt” matcht alte und neue Schreibweise.

\d{5} Postleitzahl

\d{3,6} zwischen 3 und 6 Ziffern in Folge.

Sonderzeichen bei regulären Ausdrücken

| oder

cat|dog

A cat and a dog are expected to follow the dogma that their presence with one another leads to a catastrophe.

\bcat|dog\b

A cat and a dog are expected to follow the dogma that their presence with one another leads to a catastrophe.

\b(cat|dog)\b

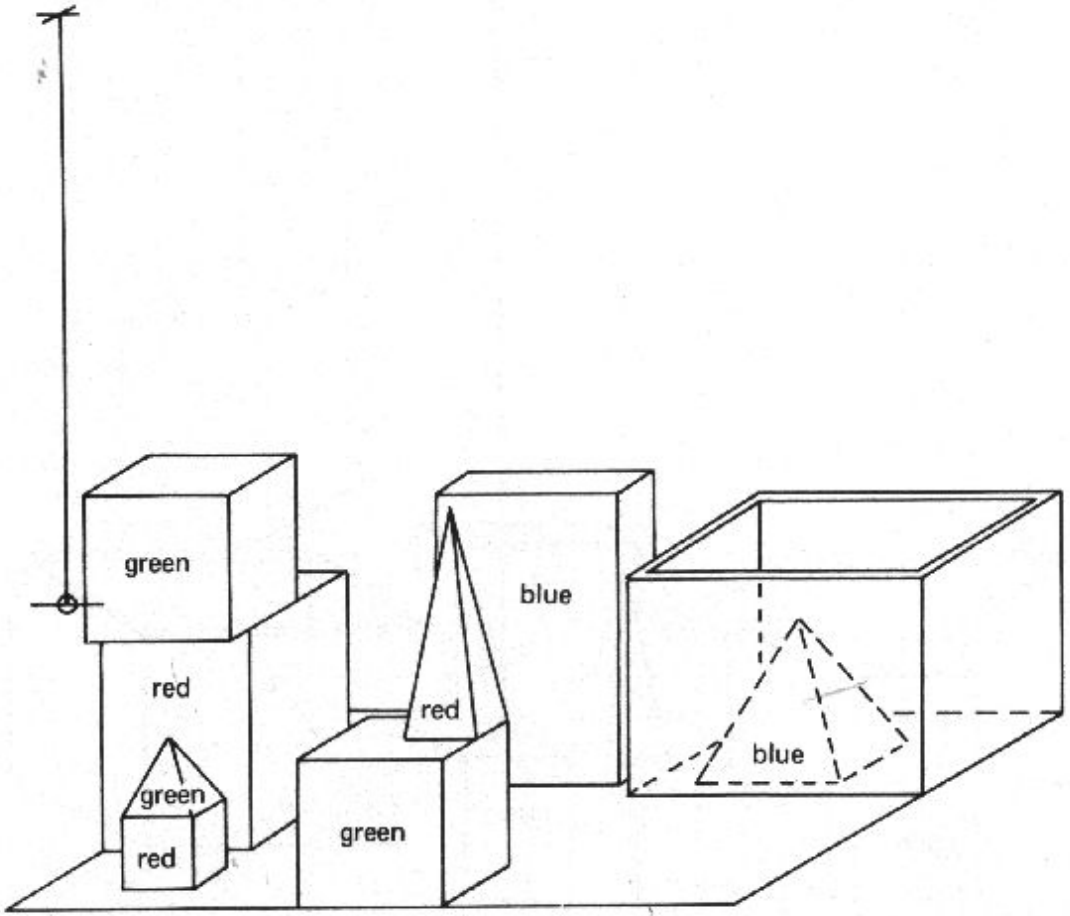
A cat and a dog are expected to follow the dogma that their presence with one another leads to a catastrophe.

ELIZA: Ein sprachverstehendes System?

- ELIZA ist ein Dialogsystem, das beliebig komplexe Eingaben mit beliebigem Wortschatz zu beliebigen Themen akzeptiert.
- ELIZA arbeitet mit einfachen Mustervergleichs-Techniken (Pattern Matching), ohne Einsatz von Wissen:
  - Templates: Muster mit variablen Teilen, die mit der Benutzereingabe abgeglichen werden, und
  - Template-basierten System-Äußerungen (Prompts)
- ELIZA hat in gewisser Hinsicht den Turing-Test absolviert, dies aber unter besonderen Rahmenbedingungen.

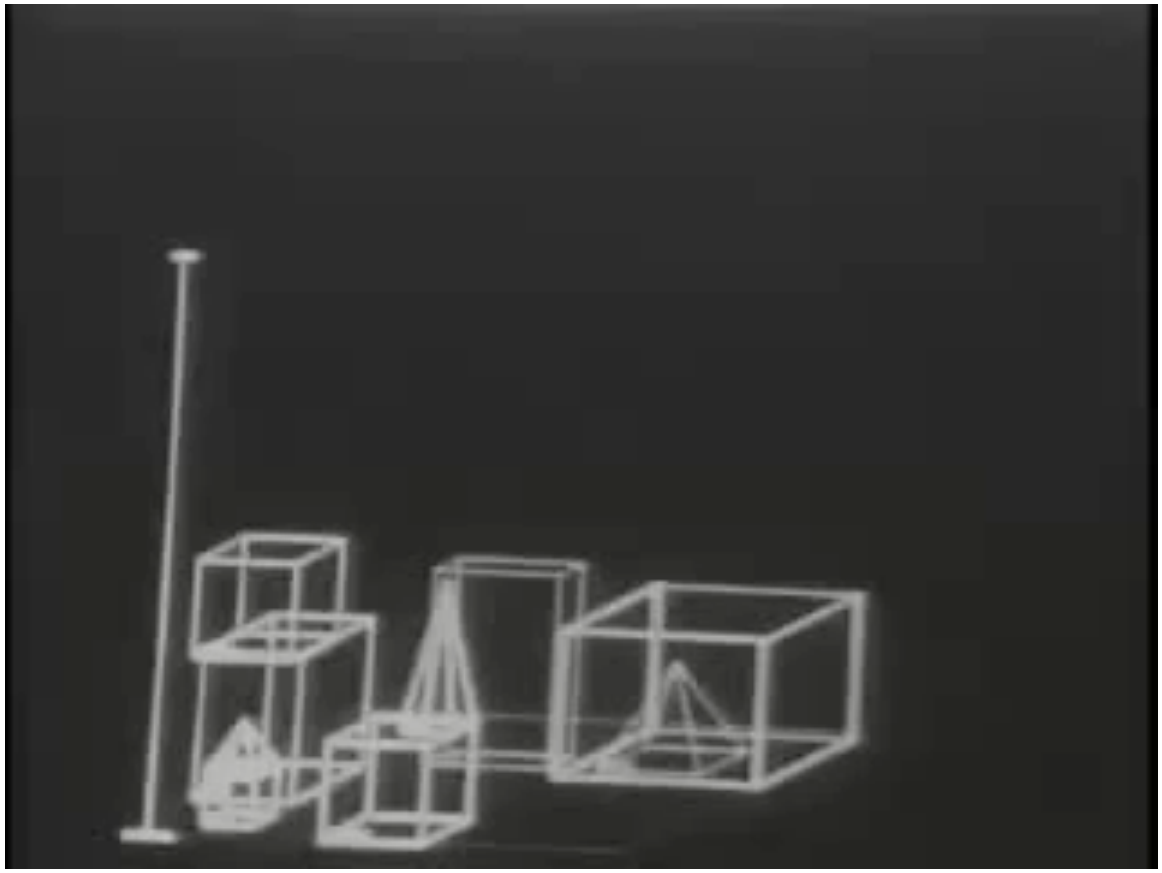
- ELIZA funktioniert besonders gut mit englischem Dialog und dem Psychotherapie-Szenario. Wieso?

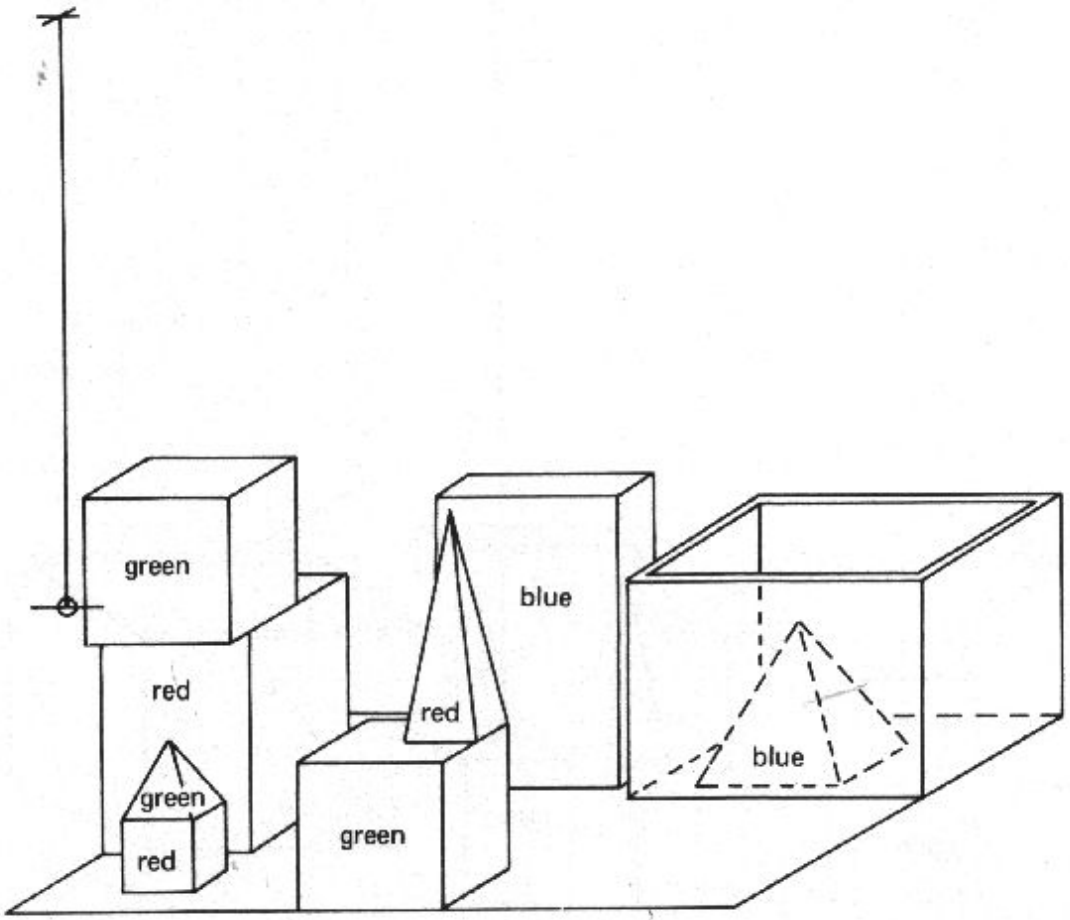
SHRDLU: Ein wissensbasiertes Dialogsystem



Winograds “Blocks World”

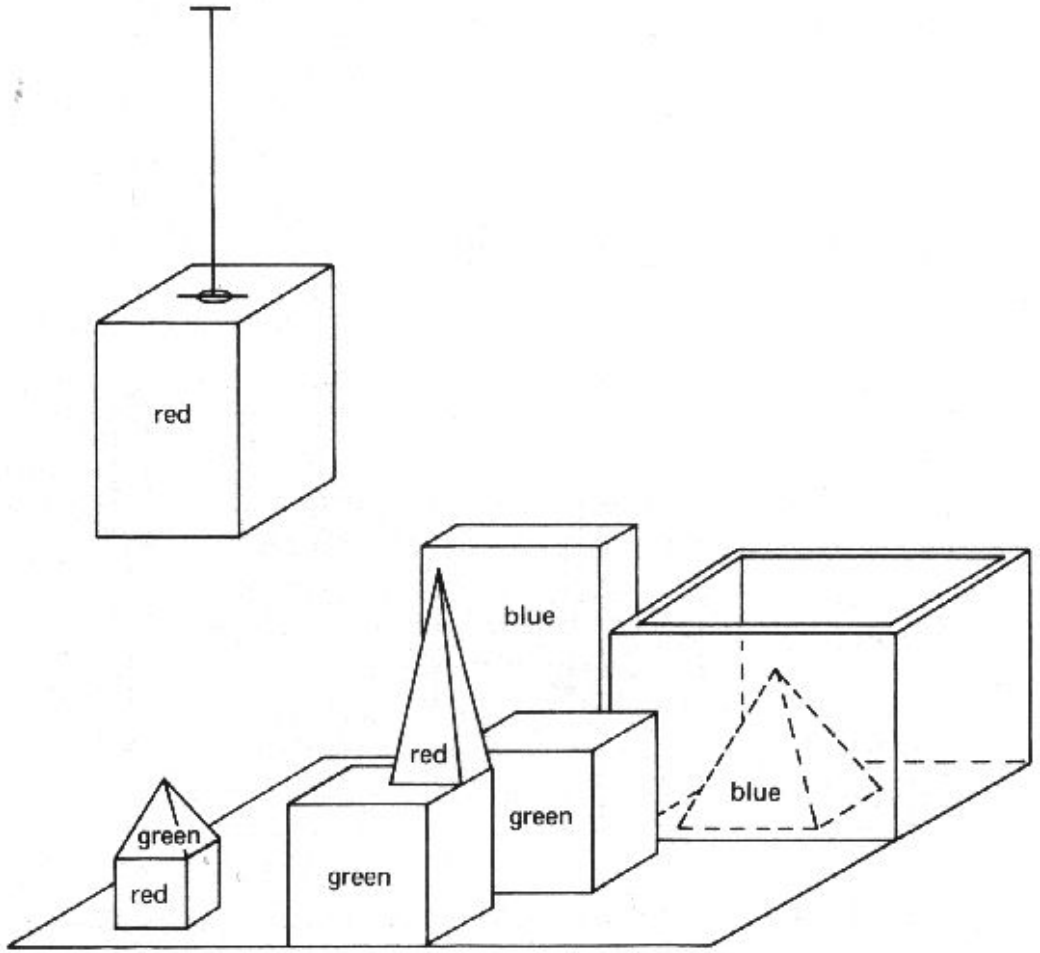
SHRDLU System (1970)



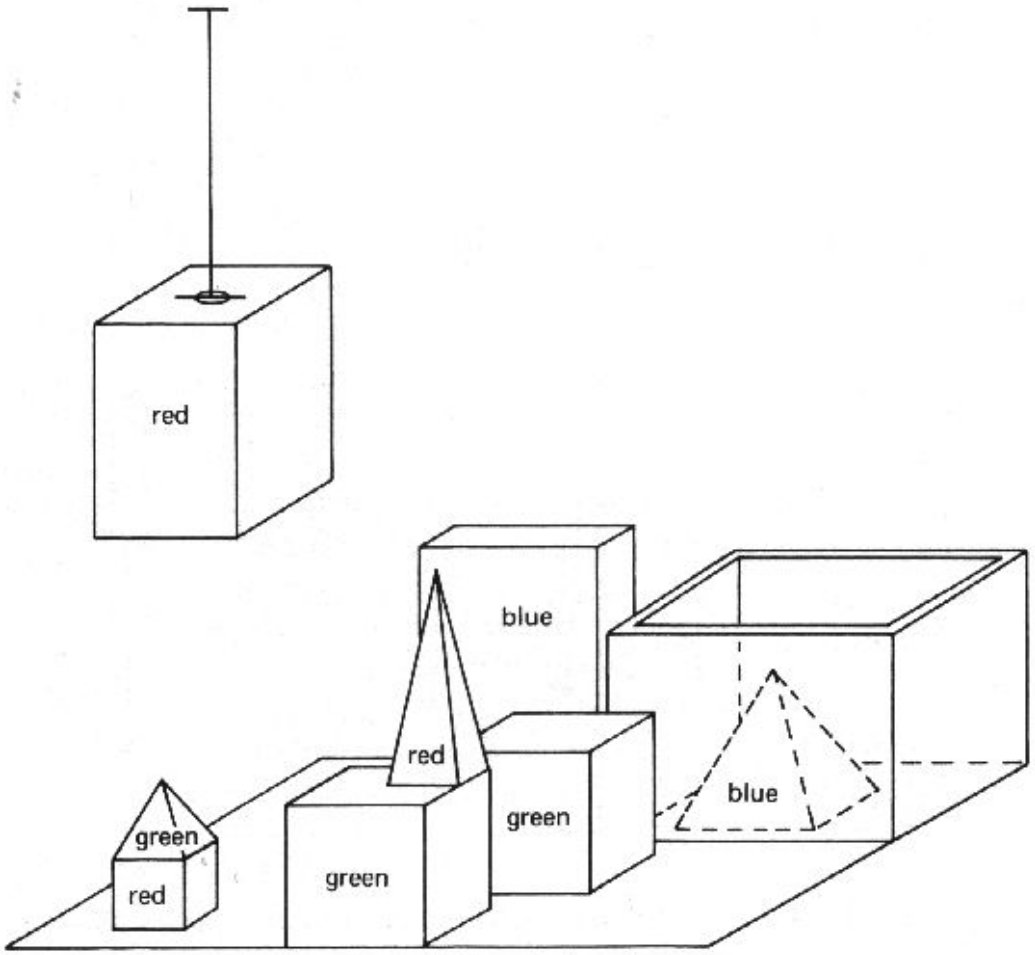


U: Pick up a big red block

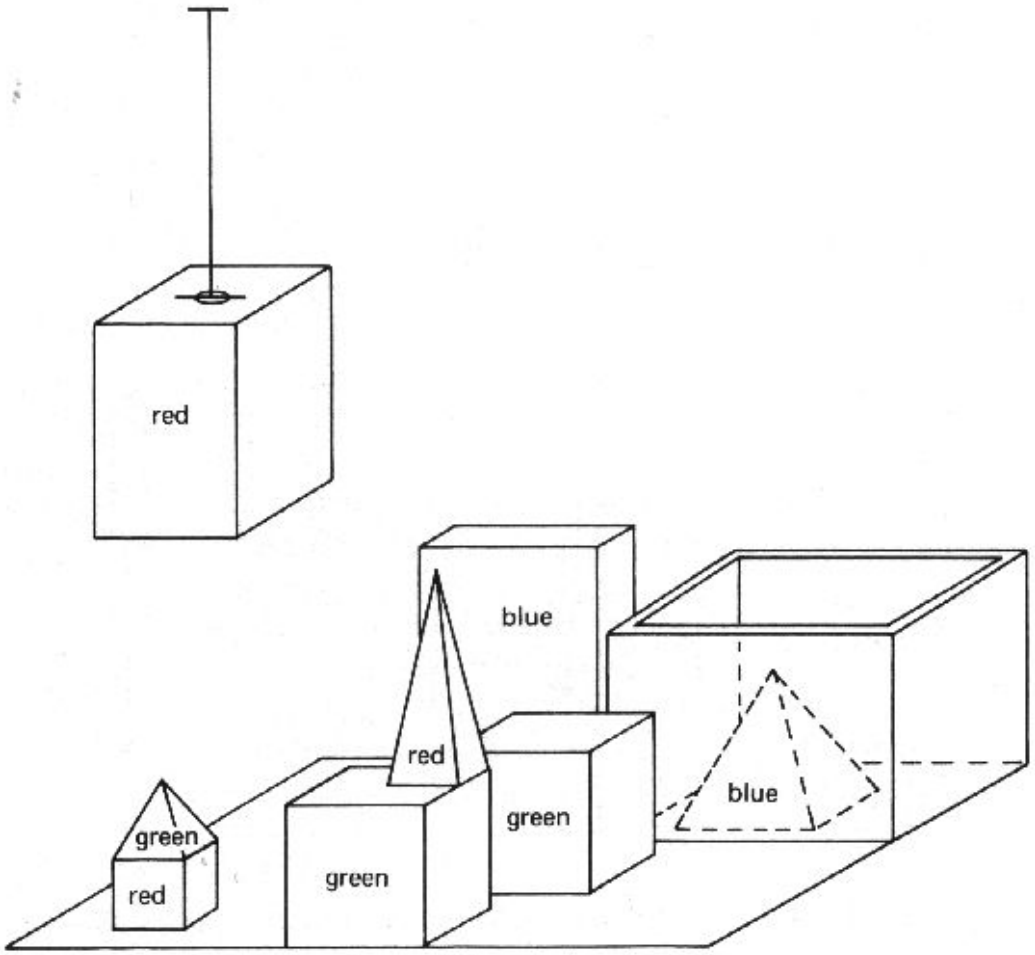
S: OK.



(Pick up a big red block)

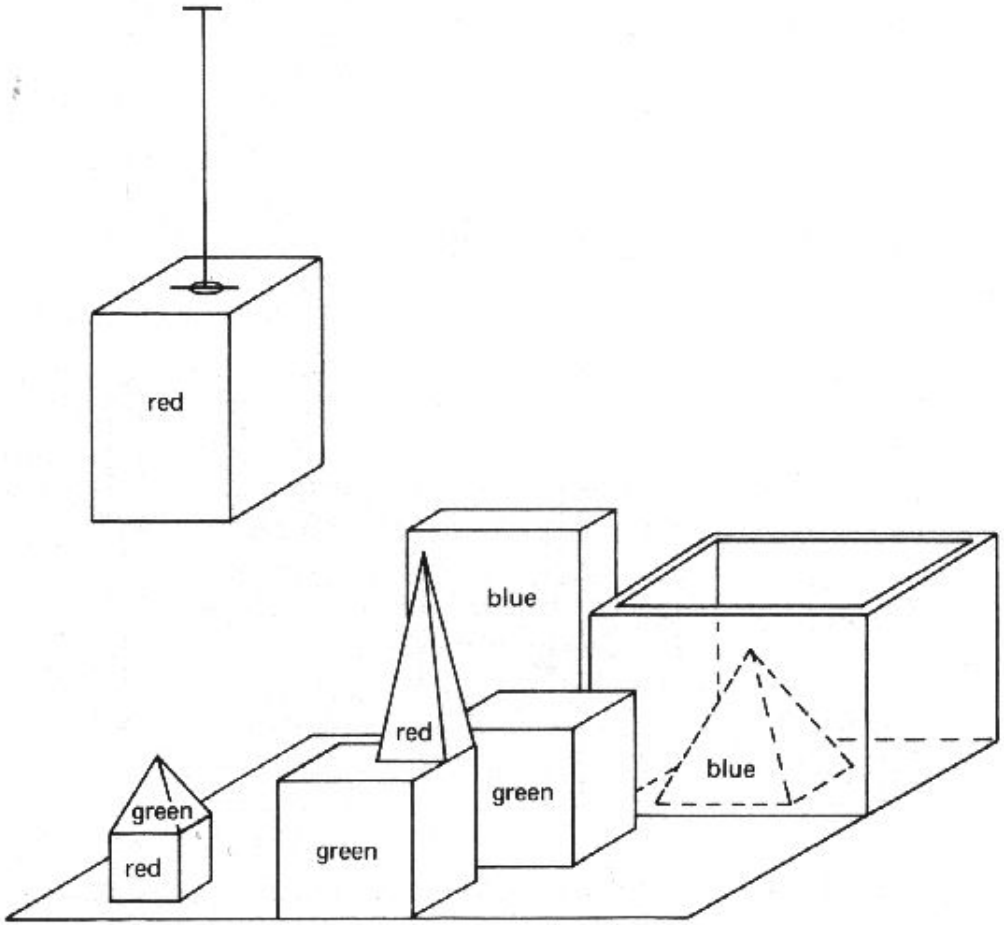


U: Grasp the pyramid.



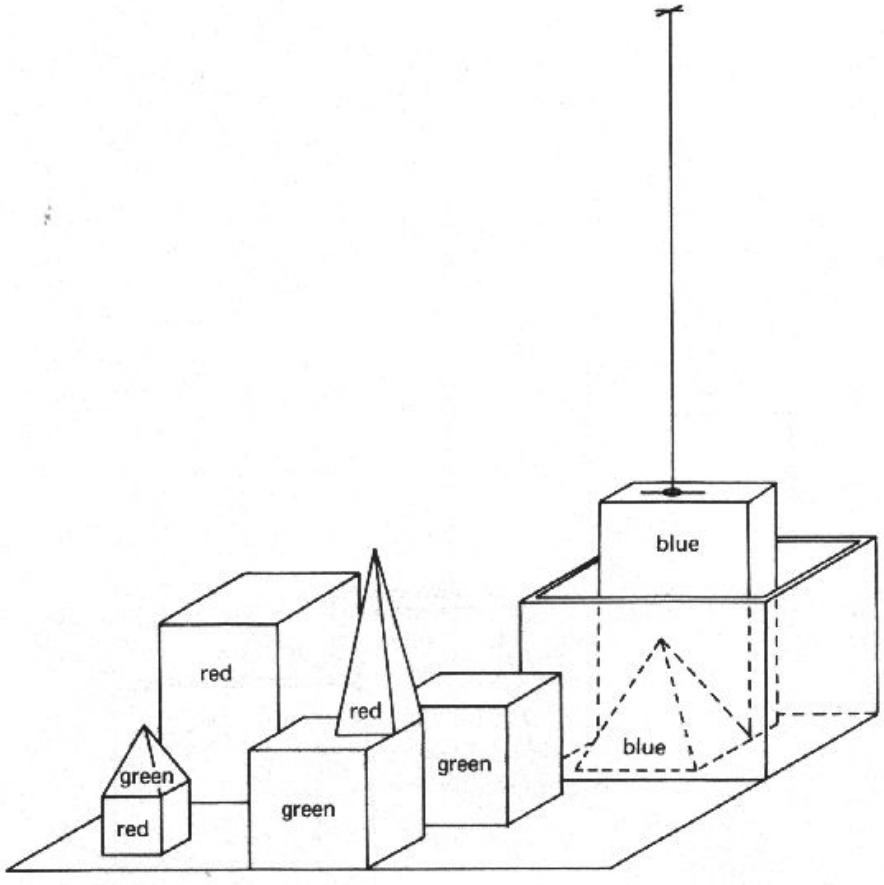
U: Grasp the pyramid.

S: I don't understand which pyramid you mean.

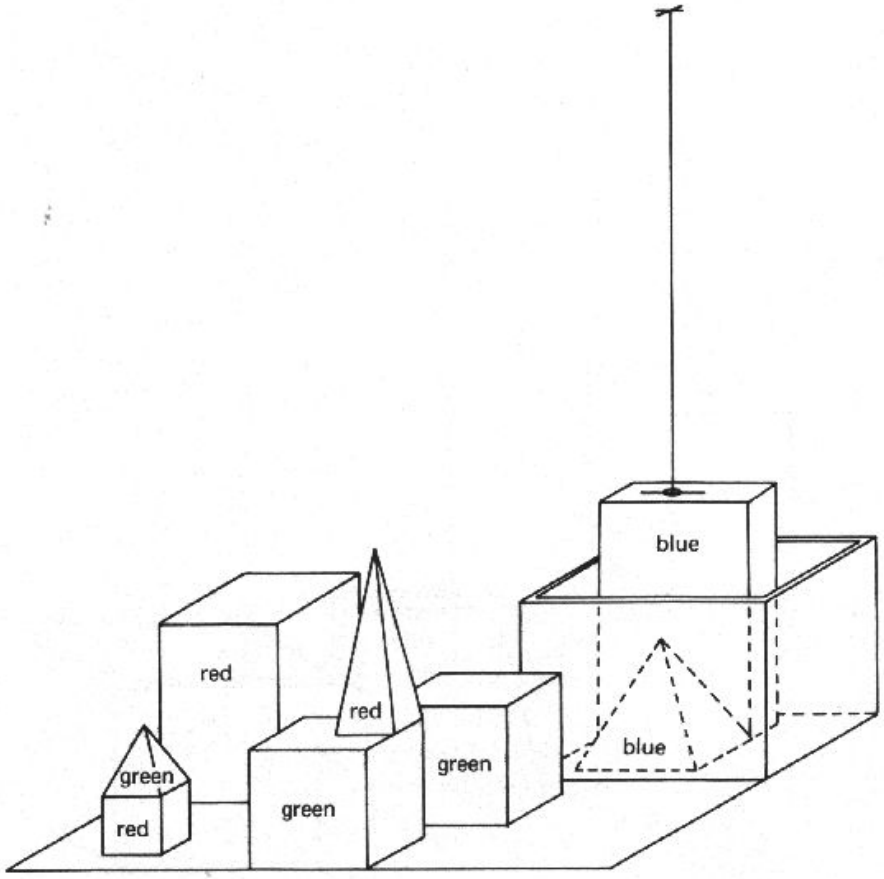


U: Find a block which is taller than the one you are holding and put it into the box.

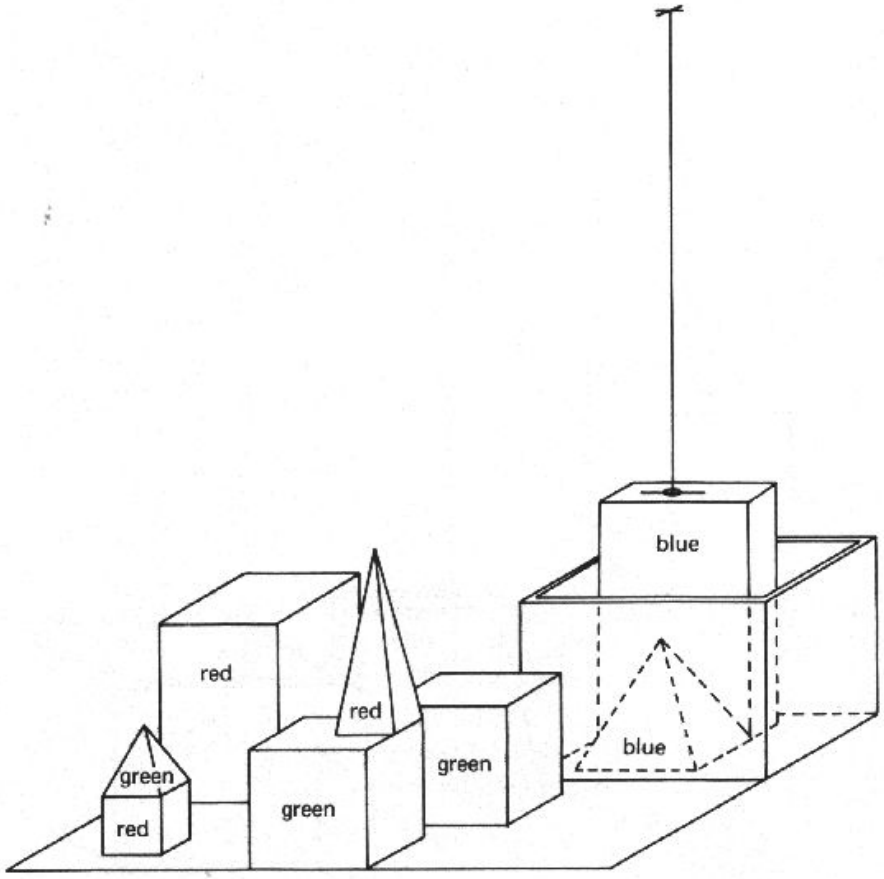
S: By „it“, I assume you mean the block which is taller than the one I am holding. OK



(U: Find a block which is taller than the one you are holding and put it into the box.)

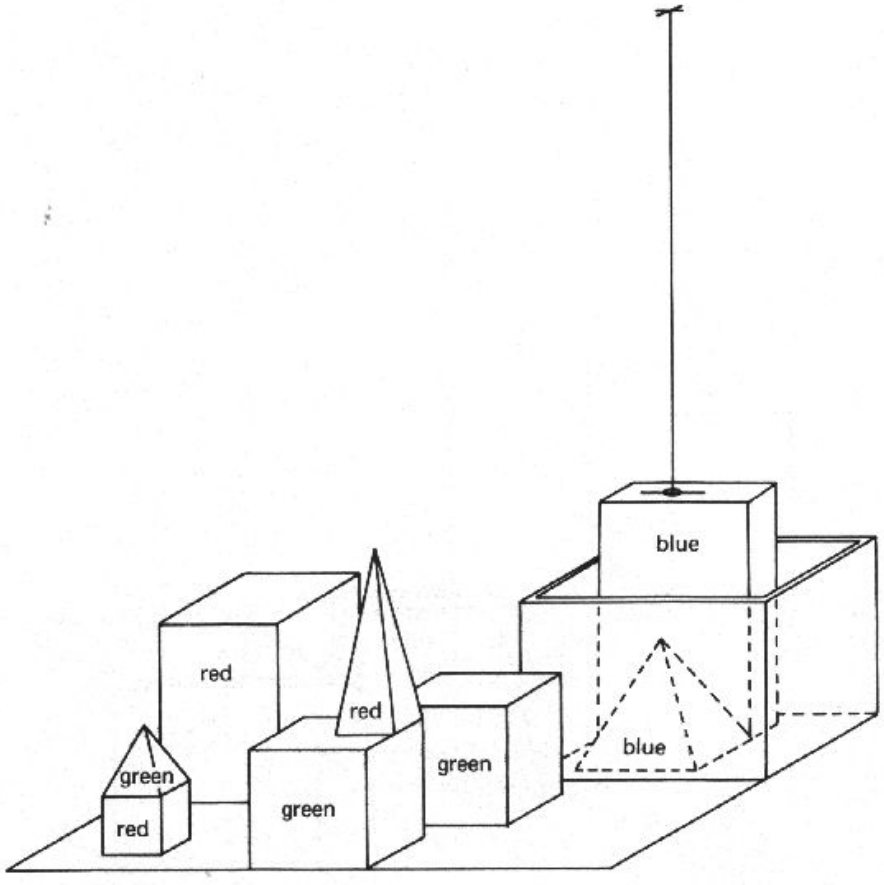


U: What does the box contain?
S: *The blue pyramid and the blue block*



U: What is the pyramid supported by?

S: The box



U: How many blocks are not in the box?
S: *Four of them*

SHRDLU

SHRDLU answers questions, executes commands and can learn new words.

Important components of SHRDLU:

- (Linguistic) analysis
- Generation
- Action planning
- Graphic visualisation

Winograd‘s SHRDLU system works in a small constrained mini domain ("Blocks World").

The interaction of planning and analysis is interesting; the generation component is very simple (just template-based); the graphics component was an important factor for the success of the system.

SHRDLU [2]

SHRDLU is a knowledge-based dialog system.

It uses different types of knowledge:

- Linguistic knowledge
- Contextual knowledge
- World knowledge

Examples for linguistic knowledge in SHRDLU

Morphological knowledge

regular verbs:	*grasp* is a regular verb
simple past ending -ed	*put* is an irregular verb (past: put)

Syntactic knowledge

imperative sentences have	*grasp* is a transitive verb
verb-first order	*stop* is an intransitive verb

Semantic knowledge

A+N in attributive	*red* denotes red objects
constructions denotes	*pyramid* is a subtype of *block*
things that are both A and N	*grasp* denotes an action, ...
N fallen	

Language-external knowledge

- Contextual knowledge:
  - Linguistic context / dialogue history: What object was mentioned last? (*Put it into the box.*)
  - Situational context: Which objects are relevant for a specific utterance setting? (*What is the pyramid supported by?*)

- World knowledge:
  - Episodic knowledge: Knowledge about specific facts

    "Es gibt zwei rote Klötze."

    "Die Kiste enthält eine Pyramide"
  - Knowledge about rules: mathematical rules, physics, social rules and norms

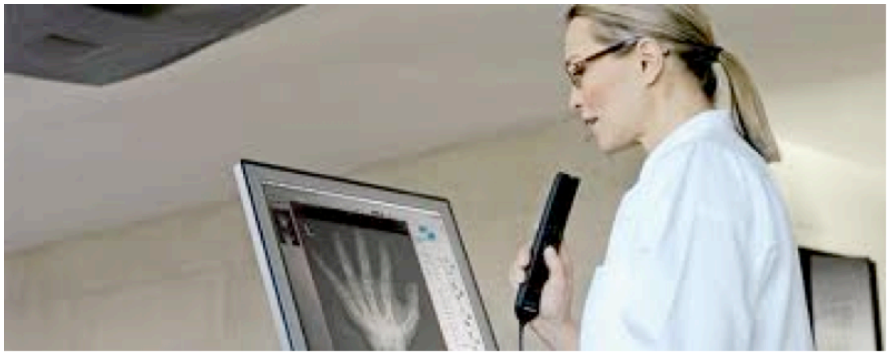
    "Zwei Objekte können nicht den gleichen Platz einnehmen."

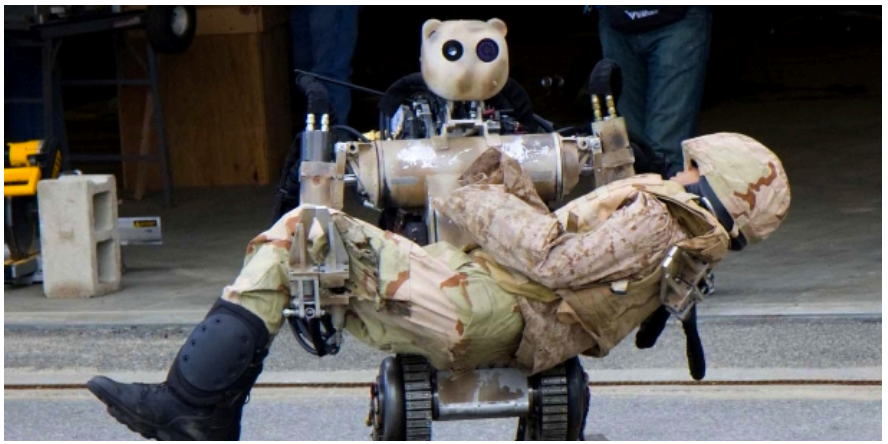
    "Ein Objekt muss eine ebene Auflagefläche besitzen, damit ein zweites stabil darauf stehen kann"

Modern dialog systems

Reasons for using dialog systems

- Hands-free operation of systems:



Reasons for using dialog systems

- Operating computer systems in a way that's intuitive to humans

Challenges in realistic dialogs

- Grounding
- Incremental processing
- Adaptation to different users

Grounding: a simple example

Dialog for operating a lift.

User:	Elevator
System:	What floor would you like to go to?
User:	Fourth floor
System:	<executes command>

Grounding: a simple example

Dialog for operating a lift.

User:	Elevator
System:	What floor would you like to go to?
User:	Fourth floor
System:	I will take you to the fourth floor.
System:	<executes command>

Grounding: a simple example

Dialog for operating a lift.

User:	Elevator
System:	What floor would you like to go to?
User:	Fourth floor
System:	I will take you to the third floor.
User:	No, to the fourth floor!
System:	I will take you to the fourth floor.
System:	<executes command>

Grounding: a simple example

Dialog for operating a lift.

User:	Elevator
System:	What floor would you like to go to?
User:	Fourth floor
System:	I didn't understand you properly. Did you say third floor?
User:	No, to the fourth floor!
System:	I will take you to the fourth floor.
System:	<executes command>

Grounding ("Verständigungssicherung")

Grounding:

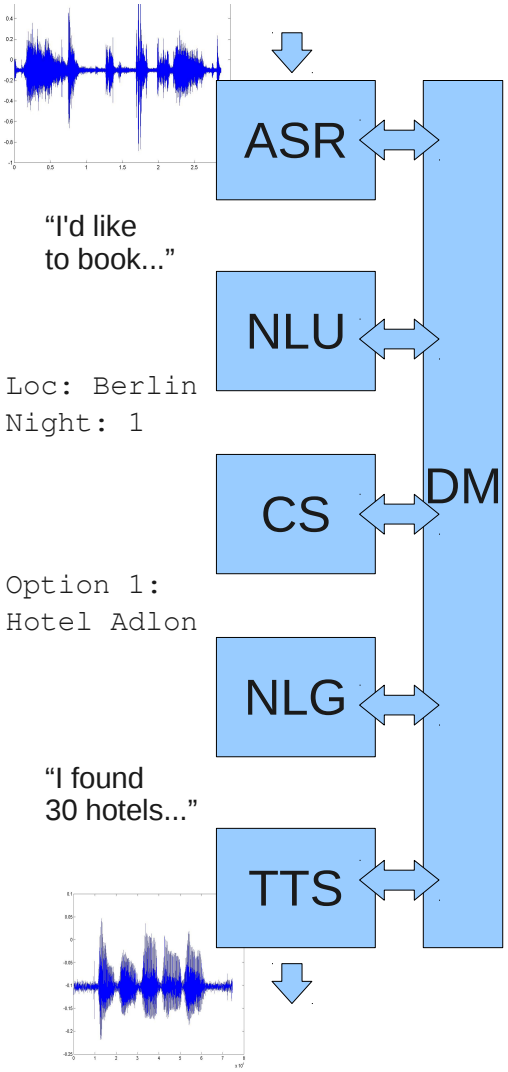
- System can ask clarification questions (explicit grounding)
- Implicit grounding, which gives users a chance to react.

Optimistic strategy: don't do grounding, hope all goes well.
Inefficient if there are misunderstandings

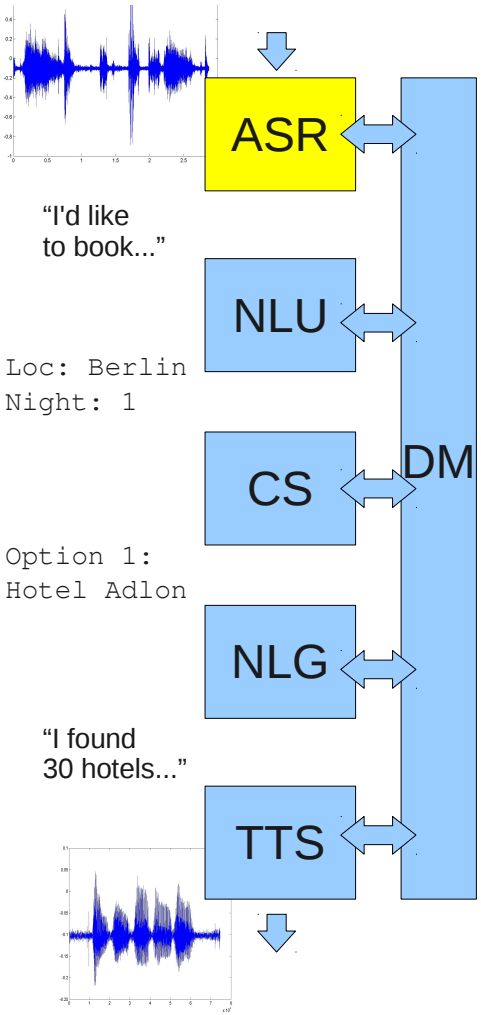
Careful strategy: constant explicit grounding; can be annoying to users, inefficient.

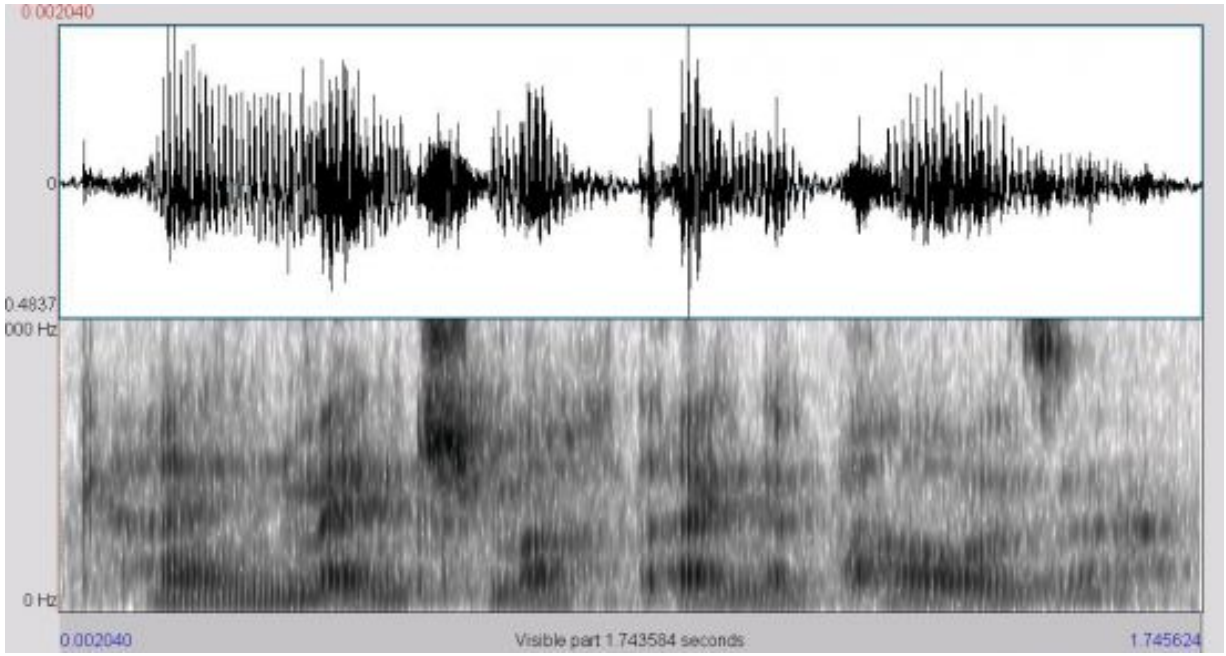
Flexible strategy: make reaction dependent on confidence: no grounding / implicit grounding / explicit clarification question.

Components of a Dialog System



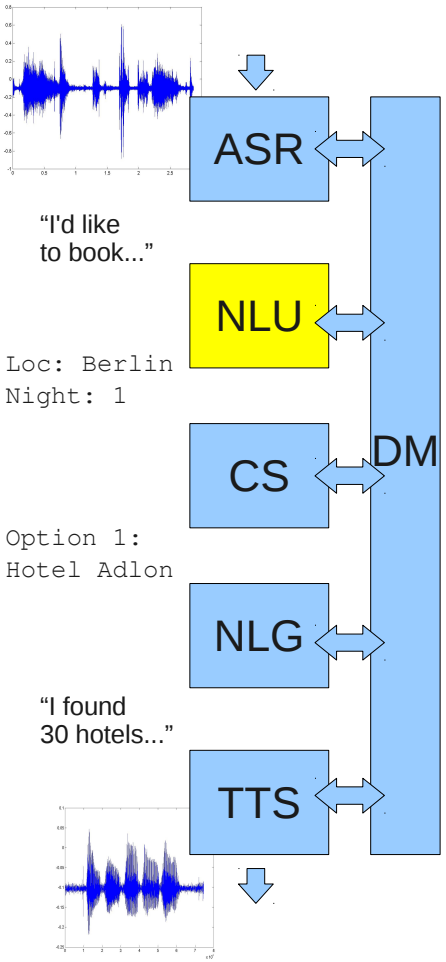
Automatic Speech Recognition





Kein Mensch macht eine Pause.

Natural Language Understanding



Example for a hotel booking system

ASR output

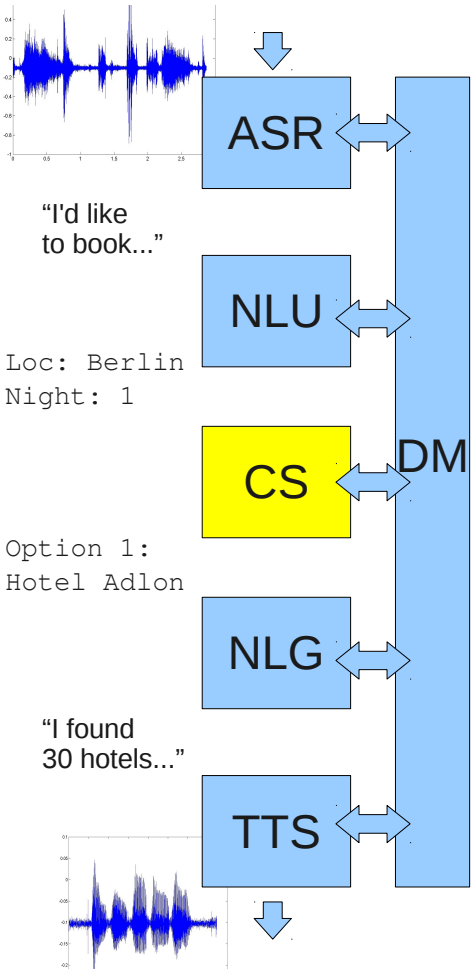
I'd like to book a double room for 2 adults in Berlin for one night.

→

Frame

city:	Berlin
arrival:	
departure:	
duration:	1
# of people:	2
room type:	double
area:	Grunewald

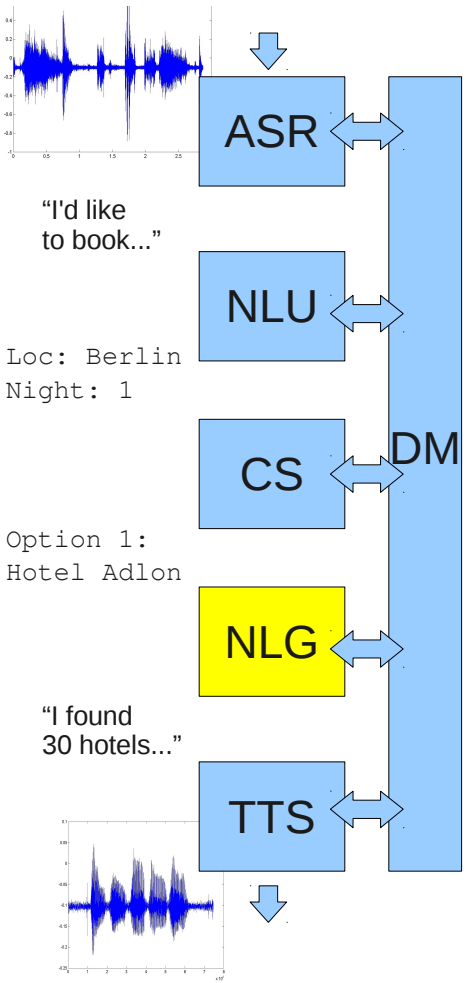
Content Selection



Select and structure contents:

1. Retrieve relevant information from data base
2. Select information that should be presented to the user
3. Structure the information (decide on order)

Textgenerierung

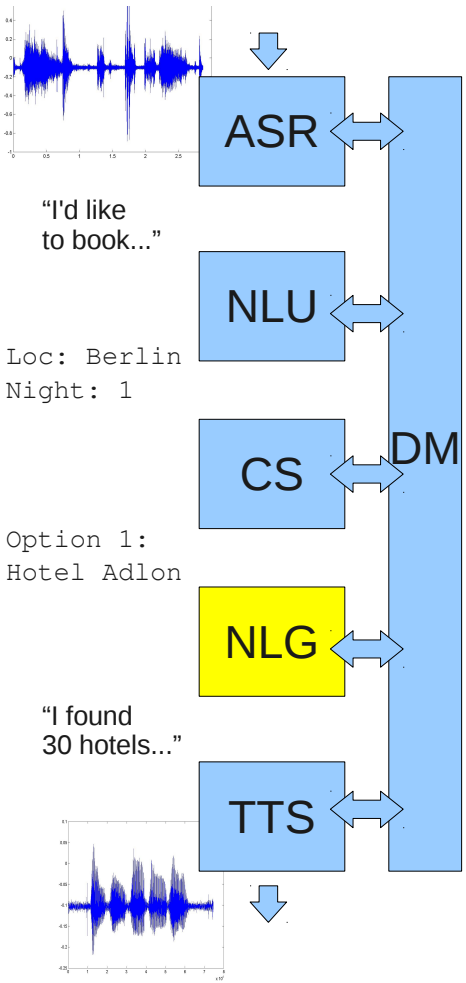


Sets of sentence templates, e.g.,

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
```

Textgenerierung



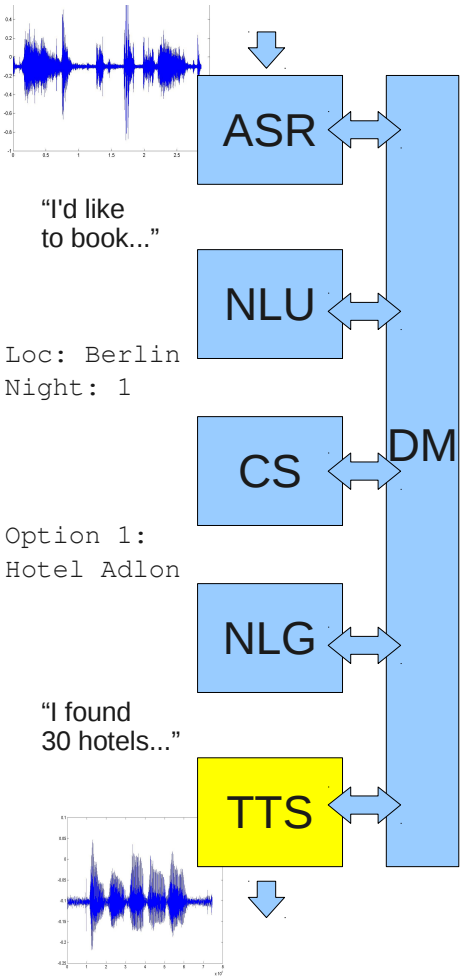
The Result:

The month was cooler than average.
The month was drier than average.
There was the average number of rainy 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.

What we'd really like:

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

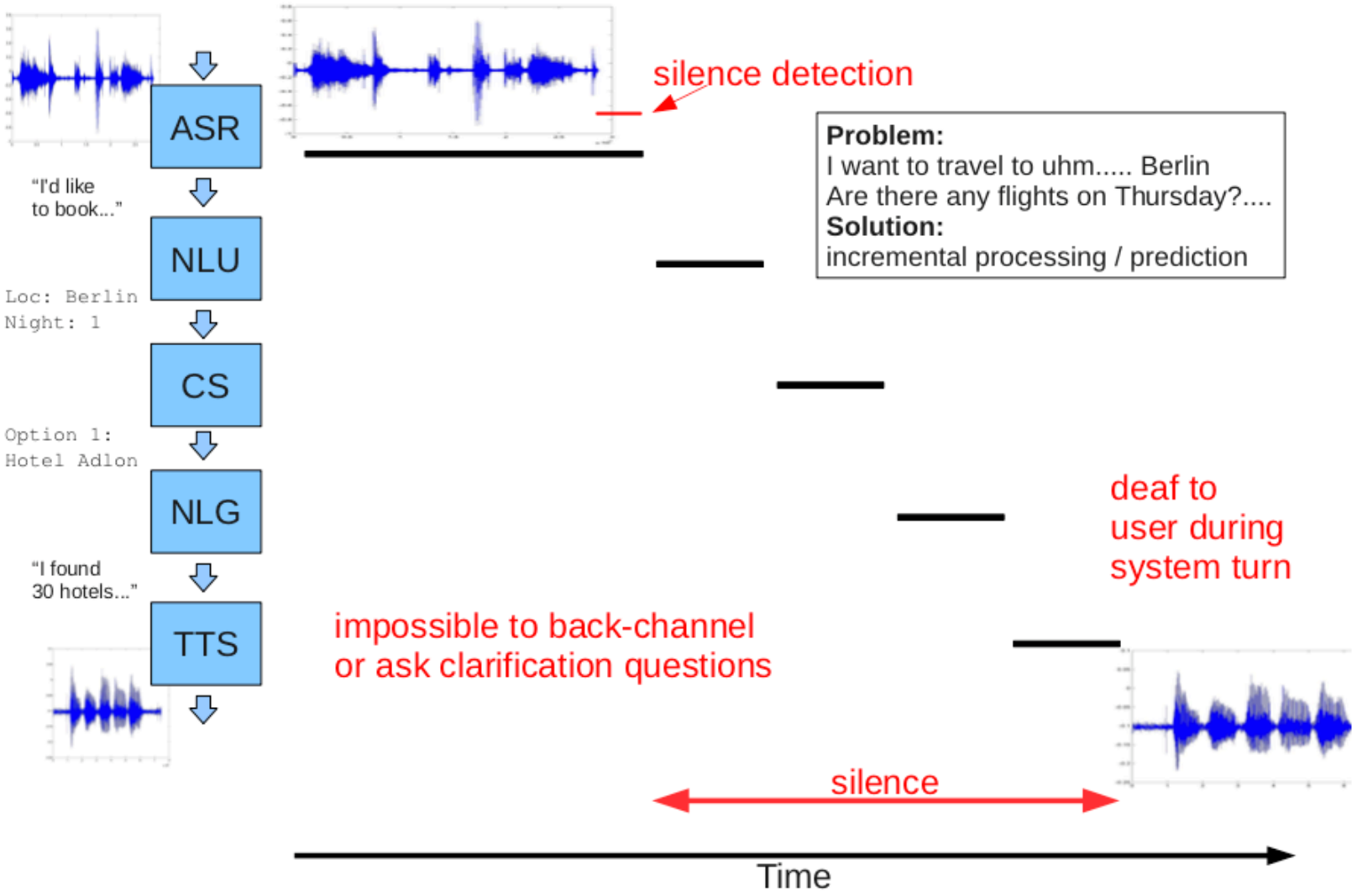
Speech synthesis



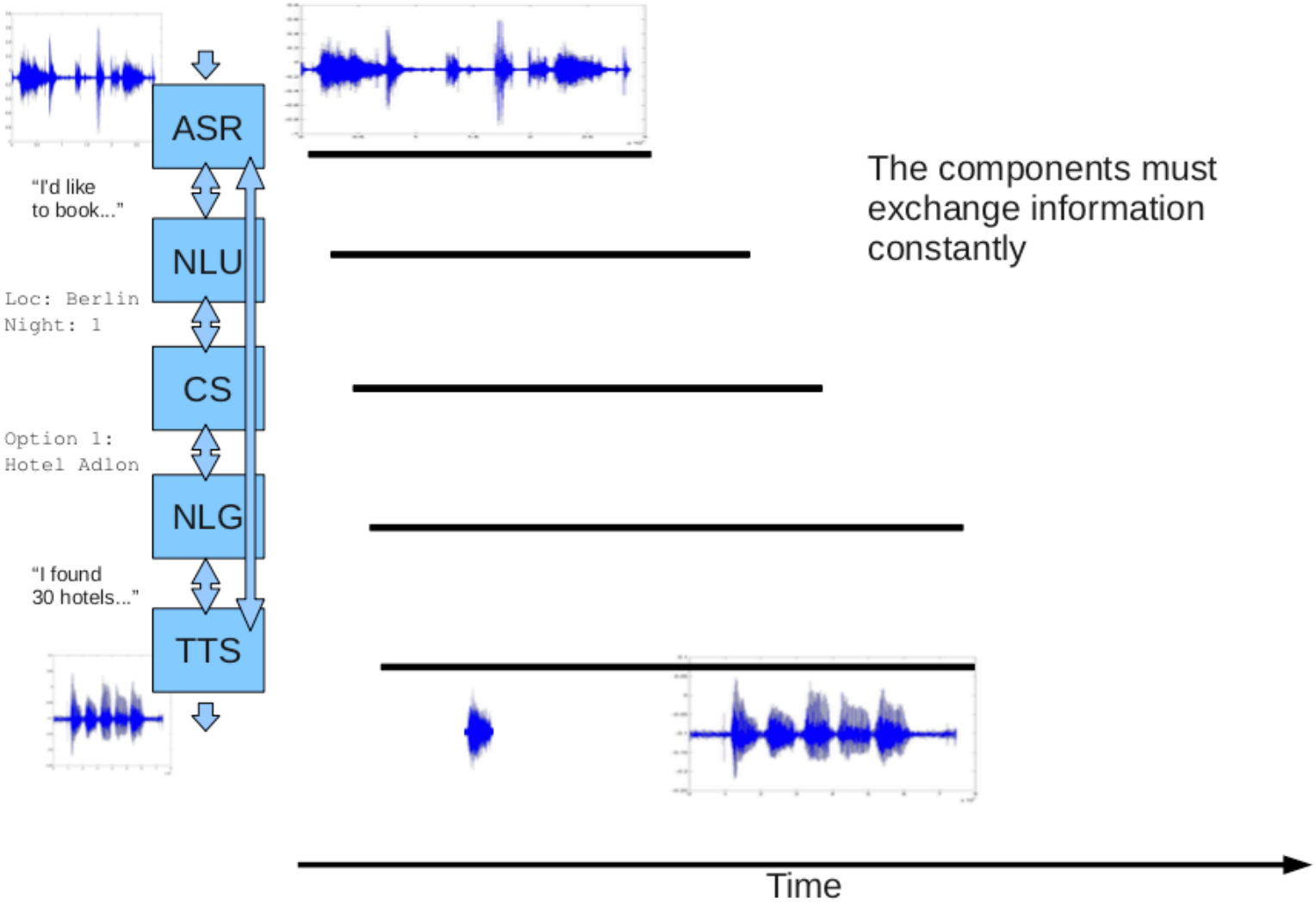
How should a word be pronounced?

- Lesart
- modern
- Aktentaschen
- 1/4
- Staubecken

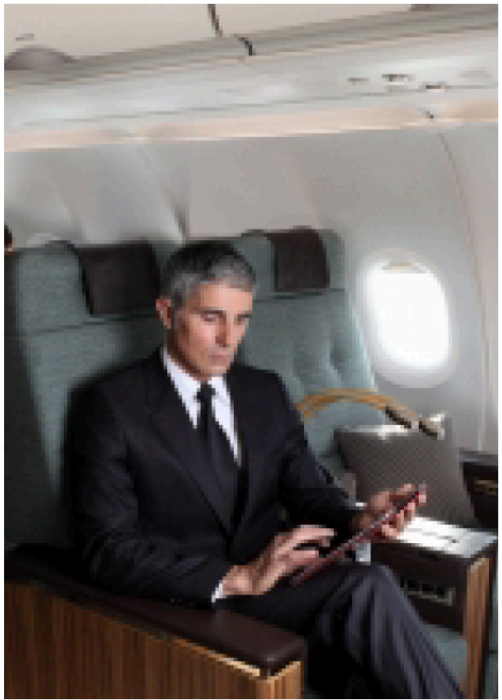
Non-incremental dialog system



Incremental System



Incremental vs. Non-incremental System (2014)



User Adaptation

Example: flight booking system

User modelling in information presentation:

- Content selection: choose options such that they are relevant for the target user.
- Point out those aspects of a flight that are attractive to the user.

Evaluation shows:

- Better user satisfaction
- More efficient interaction
- Optimal option is more likely to be chosen

Example for adaptation to user

USER: I want to travel from Edinburgh to Brussels, arriving by 5 pm.

For a starving student:

SYS: *There's a direct flight on BMI with a good price. It arrives at four ten p.m. and costs one hundred and twelve pounds. The cheapest flight is on Ryanair. It arrives at twelve forty five p.m. and costs just fifty pounds, but it requires a connection in Dublin.*

For a business traveller:

SYS: *You can fly business class on British Airways, arriving at four twenty p.m., but youd need to connect in Manchester. There's a direct flight on BMI, arriving at four ten p.m., but theres no availability in business class.*

Figure: Beispiel FLIGHTS system; White et al., 2010

Natural Language Processing at UdS

Prof. Dr. Matthew Crocker
Psycholinguistics

Prof. Dr. Vera Demberg
Discourse, Generation, Cognitive Modelling

Prof. Dr. Josef van Genabith
Machine Translation

Prof. Dr. Dietrich Klakow
Statistical / Neural NLP, Dialog Systems

Prof. Dr. Alexander Koller
Parsing, Generation, Dialog

Prof. Dr. Bernd Möbius
Phonetics & Phonology

Prof. Dr. Elke Teich
Corpus Linguistics, English Translation

Jun.Prof. Dr. Annemarie Verkerk
Language Science / Typology