

# Language Technology

<http://cs.lth.se/edan20/>  
Chapter 1: An Overview of Language Processing

Pierre Nugues

Pierre.Nugues@cs.lth.se  
[http://cs.lth.se/pierre\\_nugues/](http://cs.lth.se/pierre_nugues/)

August 30, 2021



# Applications of Language Processing

- Spelling and grammatical checkers: *MS Word*, e-mail programs, etc.
- Text indexing and information retrieval on the Internet: *Google*, *Microsoft Bing*, *Yahoo*, or software like *Apache Lucene*
- Translation: *Google Translate*, *DeepL*, *Bing translator*, etc.
- Spoken interaction: Apple Siri, Google Assistant, Amazon Echo
- Speech dictation of letters or reports: *Windows 10*, *macOS*



# Applications of Language Processing (ctn'd)

- Direct translation from spoken English to spoken Swedish in a restricted domain: *SRI* and *SICS*
- Voice control of domestic devices such as tape recorders: *Philips* or disc changers: *MS Persona*
- Conversational agents able to dialogue and to plan: *TRAINS*
- Spoken navigation in virtual worlds: *Ulysse*, *Higgins*
- Generation of 3D scenes from text: *Carsim*
- Question answering: *IBM Watson* and *Jeopardy!*



# Linguistics Layers

- Sounds
- Phonemes
- Words and morphology
- Syntax and functions
- Semantics
- Dialogue



# Sounds and Phonemes



*Serious*



*C'est par là* 'It is that way'



# Lexicon and Parts of Speech

*The big cat ate the gray mouse*

*The/article big/adjective cat/noun ate/verb the/article gray/adjective mouse/noun*

*Le/article gros/adjectif chat/nom mange/verbe la/article souris/nom grise/adjectif*

*Die/Artikel große/Adjektiv Katze/Substantiv ißt/Verb die/Artikel graue/Adjektiv Maus/Substantiv*



# Morphology

Word	Root form
<i>worked</i>	<i>to work</i> + verb + preterit
<i>travaillé</i>	<i>travailler</i> + verb + past participle
<i>gearbeitet</i>	<i>arbeiten</i> + verb + past participle



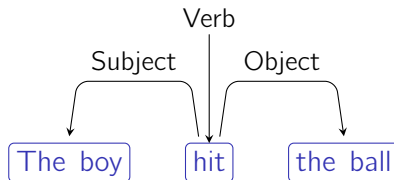
# Syntactic Tree





# Syntax: A Classical View

A graph of dependencies and functions



# Semantics

As opposed to syntax:

- ① Colorless green ideas sleep furiously.
- ② \*Furiously sleep ideas green colorless.

Determining the logical form:

Sentence	Logical representation
Frank is writing notes	writing(Frank, notes).
François écrit des notes	écrit(François, notes).
Franz schreibt Notizen	schreibt(Franz, Notizen).



# Lexical Semantics

Word senses:

- ① **note** (*noun*) short piece of writing;
- ② **note** (*noun*) a single sound at a particular level;
- ③ **note** (*noun*) a piece of paper money;
- ④ **note** (*verb*) to take notice of;
- ⑤ **note** (*noun*) of note: of importance.



# Reference

## 1. Sentence

*Pierre wrote notes*

## 2. Logical representation

`wrote(pierre, notes)`

## 3. Real world

Louis



Pierre

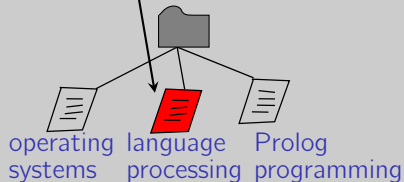


Charlotte



refers to

refers to



# Ambiguity

Many analyses are ambiguous. It makes language processing difficult. Ambiguity occurs in any layer: speech recognition, part-of-speech tagging, parsing, etc.

Example of an ambiguous phonetic transcription:

*The boys eat the sandwiches*

That may correspond to:

*The boy seat the sandwiches; the boy seat this and which is; the buoys eat the sand which is*



# Models and Tools

- Linguistics has produced an impressive set of theories and models;
- Inadequate theories in the beginning and lack of data: corpus, dictionaries, or reference (annotated) data;
- Models and tools have matured. Data has become available;
- Tools involve notably finite-state automata, regular expressions, logic, statistics, and machine learning;
- In general, language processing requires significant processing power;
- This overall resulted in massive improvements in most areas of NLP.



# The Carsim System: A Text-to-Scene Converter

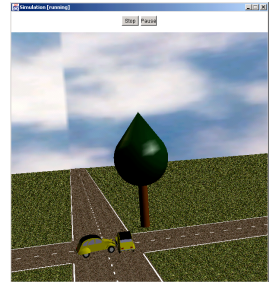
Texts

XML Templates

3D Animation

*Véhicule B venant de ma gauche, je me trouve dans le carrefour, à faible vitesse environ 40 km/h, quand le véhicule B, percute mon véhicule, et me refuse la priorité à droite. Le premier choc atteint mon aile arrière gauche,*

```
// Static Objects
STATIC [
ROAD
TREE
]
// Dynamic Objects
DYNAMIC [
VEHICLE [
ID = vehicule_b;
INITDIRECTION = east;
```



# Dialogue: The Persona Project from Microsoft Research

## A conversation with Peedy

Turn	Utterance
	[Peedy is asleep on his perch]
User:	Good morning, Peedy.
	[Peedy rouses]
Peedy:	Good morning.
User:	Let's do a demo.
	[Peedy stands up, smiles]
Peedy:	Your wish is my command, what would you like to hear?
User:	What have you got by Bonnie Raitt?
	[Peedy waves in a stream of notes, and grabs one as they rush by.]
Peedy:	I have "The Bonnie Raitt Collection" from 1990.
User:	Pick something from that
Peedy:	How about "Angel from Montgomery"?





# Dialogue: The Persona Project from Microsoft Research

User: Sounds good.

[Peedy drops note on pile]

Peedy: OK.

User: Play some rock after that.

[Peedy scans the notes again, selects one]

Peedy: How about "Fools in love"?

User: Who wrote that?

[Peedy cups one wing to his 'ear']

Peedy: Huh?

User: Who wrote that?

[Peedy looks up, scrunches his brow]

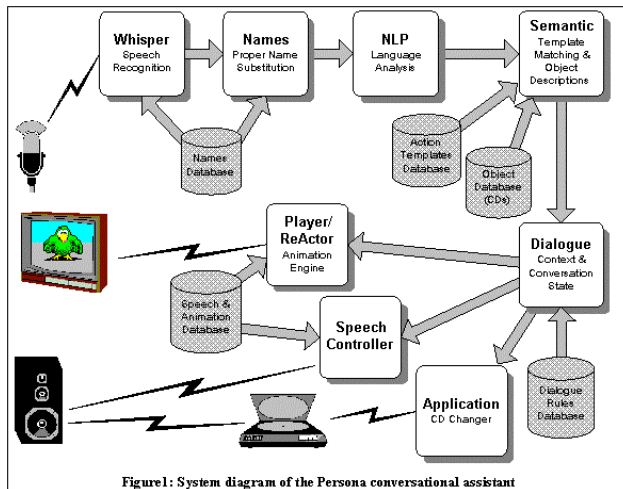
Peedy: Joe Jackson

User: Fine.

[Drops note on pile]



# Persona System Architecture



Source: <http://research.microsoft.com/research/pubs/view.aspx?pubid=439>

<http://research.microsoft.com/research/pubs/view.aspx?pubid=439>

# IBM Watson

- IBM Watson: A system that can answer questions better than any human
- Video:  
[https://www.youtube.com/watch?v=WFR310m\\_xhE](https://www.youtube.com/watch?v=WFR310m_xhE)



- IBM Watson builds on the extraction of knowledge from masses of texts: Wikipedia, archive of the New York Times, etc.
- Bottom line: Text is the repository of human knowledge



# IBM Watson: Simplified Architecture



Question parsing and classification:

*Syntactic parsing,  
entity recognition,  
answer classification*

Document retrieval.

Extraction and ranking of passages:  
*Indexing, vector space model.*

Extraction and ranking of answers:

*Answer parsing, entity recognition*

