

# **Introduction to Computational Linguistics**

**PD Dr. Frank Richter**

**fr@sfs.uni-tuebingen.de.**

**Seminar für Sprachwissenschaft  
Eberhard-Karls-Universität Tübingen  
Germany**

# MT: Performance Google/Altavista (1)

- Maria hat dem Kind ein Buch gegeben.  
Maria gave a book to the child.

# MT: Performance Google/Altavista (1)

- Maria hat dem Kind ein Buch gegeben.  
Maria gave a book to the child.
- Ich glaube nicht, dass diese Maschine gute Übersetzungen liefern kann.  
I do not believe that this machine can supply good translations.

# MT: Performance Google/Altavista (1)

- Maria hat dem Kind ein Buch gegeben.  
Maria gave a book to the child.
- Ich glaube nicht, dass diese Maschine gute Übersetzungen liefern kann.  
I do not believe that this machine can supply good translations.
- Wenn man einen Satz aus der Zeitung nimmt, dann müßte das Programm ihn übersetzen können.  
If one takes a sentence from the newspaper, then the program would have to be able to translate him.

# MT: Performance Google/Altavista (2)

- Peter hat den Löffel abgegeben.  
Peter delivered the spoon.

# MT: Performance Google/Altavista (2)

- Peter hat den Löffel abgegeben.  
Peter delivered the spoon.
- Das ist nicht der Grund dafür, dass ich ihm nicht traue.  
That is not the reason for the fact that I do not trust it.

# MT Performance: An Example

In Zusammenhang mit der Eröffnung der Repraesentation in Deutschland, sucht Gesellschaft ESolutions Inc. die Mitarbeiter auf verschiedene Vakanzen.

Falls Sie sind schon aelter als 21 Jahre alt und gute Arbeit bekommen wollen, schicken Sie uns die eigene Zusammenfassung her. Wir haben unbesetzten Stellen wie fuer die Spezialisten, als auch fuer die Arbeiter ohne spezielle Fertigkeiten und die Bildungen. When Sie haben eine Interesse ueber unsere Vorschlag und moechten mehr Information bekommen so koennen Sie sich mit uns verbinden verwendend die untenangefuhrte Form.

...

# Some Misconceptions about MT (1)

- **False:** MT is a waste of time because you will never make a machine that can translate Shakespeare.



# Some Misconceptions about MT (1)

- **False:** MT is a waste of time because you will never make a machine that can translate Shakespeare.
- **False:** There was/is an MT system which translated *the spirit is willing, but the flesh is weak* into the Russian equivalent of *The vodka is good, but the steak is lousy*, and *hydraulic ram* into the French equivalent of *water goat*. MT is useless.

# Some Misconceptions about MT (2)

- **False:** Generally, the quality of translation you can get from an MT system is very low. This makes them useless in practice.

# Some Misconceptions about MT (2)

- **False:** Generally, the quality of translation you can get from an MT system is very low. This makes them useless in practice.
- **False:** MT threatens the jobs of translators.

# Some Misconceptions about MT (2)

- **False:** Generally, the quality of translation you can get from an MT system is very low. This makes them useless in practice.
- **False:** MT threatens the jobs of translators.
- **False:** The Japanese have developed a system that you can talk to on the phone. It translates whatever you say into Japanese, and translates the other speaker's replies into English.

# Some Misconceptions about MT (3)

- **False:** There is a amazing South American Indian language with a structure of such logical perfection that it solves the problem of designing MT systems.

# Some Misconceptions about MT (3)

- **False:** There is a amazing South American Indian language with a structure of such logical perfection that it solves the problem of designing MT systems.
- **False:** MT systems are machines, and buying an MT system should be very much like buying a car.

# Incremental Linguistic Analysis

- tokenization
- morphological analysis (lemmatization)
- part-of-speech tagging
- named-entity recognition
- partial chunk parsing
- full syntactic parsing
- semantic and discourse processing

# Tokenization: Motivation

- Robust NLP
- Processing of large corpora
- Preprocessing step for other applications



# Preprocessing the Text: Tokenization

Tokenization refers to the annotation step of dividing the input text into units called *tokens*.

Each token consists of either:

- a morpho-syntactic word
- a punctuation mark or a special character (e.g. &, @, %)
- a number

# Why is Tokenization Non-trivial?

- Disambiguation of punctuation

E.g. period can occur inside cardinal numbers, after ordinals, after abbreviations, at end of sentences

- Recognition of complex words

- compounds, e.g. *bank transfer fee*, *US-company*
- mergers, e.g. cliticization in French *t'aime* or English *England's*
- multiwords, e.g. complex prepositions (*provided that*, *in spite of*)

# Tokenization for Japanese

Japanese: the ultimate nightmare for tokenization!

Just take a look: e.g.

<http://www.yomiuri.co.jp/index-j.htm>

What is so hard ?

- cannot rely on blank spaces and punctuation
- combination of two writing systems: kanji (Chinese characters) and hiragana (mostly used for marking grammatical endings)
- E.g. WATASHI-wa (“first party”, meaning: I); large cap part is in Kanji and remaining part is in hiragana.

# Deterministic Tokenization

- If the output never contains alternative segmentations for any part of the input, the tokenizer is called **deterministic**.
- Deterministic tokenization is commonly seen as an independent preprocessing step unambiguously producing items for subsequent morphological analysis.
- Deterministic tokenization is commonly implemented as a cascade of finite-state transducers.