Introduction to Computational Linguistics

General Introduction, History of CL

Jan-Philipp Söhn

jp.soehn@uni-tuebingen.de

October 24th, 2007

Course Goals

- First (largely non-technical) introduction to the field of computational linguistics and its history.
- Survey of natural language processing applications
- In-depth look at machine translation as a means to illustrate the major tasks for natural language processing
- Presentation of tools and resources needed for natural language processing applications

The Name of the Field

- Computational Linguistics
- Natural Language Processing
- Human Language Technology
- Language Engineering

The History of the Field

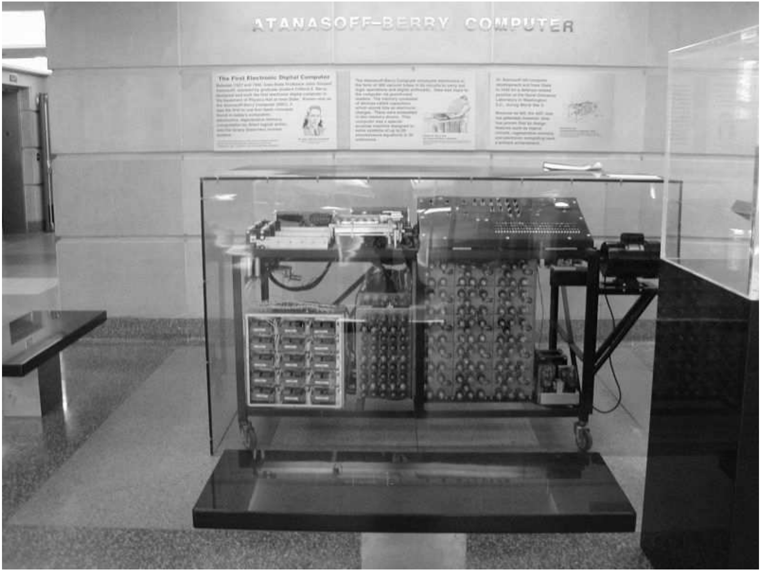
- Not surprisingly, the history of Computational Linguistics is closely connected to the development of the digital computer.
- The theoretical foundations of Computational Linguistics are in the field of mathematical linguistics, formal language theory, and in the field of information theory.
- The first application of Computational Linguistics was in the area of machine translation. Therefore, the first professional organization was called *Association for Machine Translation and Computational Linguistics*.

The History of the Digital Computer (1)

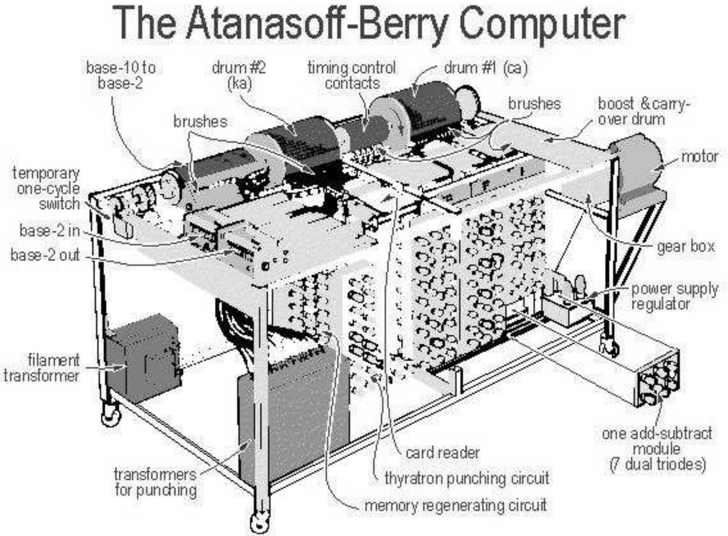
1939 John J. Atanasoff designs a prototype for the
ABC (Atanasoff-Berry Computer) with the help of graduate
graduate student Clifford Berry at Iowa State
College. In 1973 a judge ruled it the
first automatic digital computer.
320 kg; 1.6 km of wire; 280 dual-triode vacuum tubes;
30 additions/subtractions per second;

1941 Colossus computer is designed by M.H.A. Neuman
at the University of Manchester, England,
building on ideas of Alan M. Turing.
Used by British codebreakers during World War II.
1500 vacuum tubes; 4500 W;

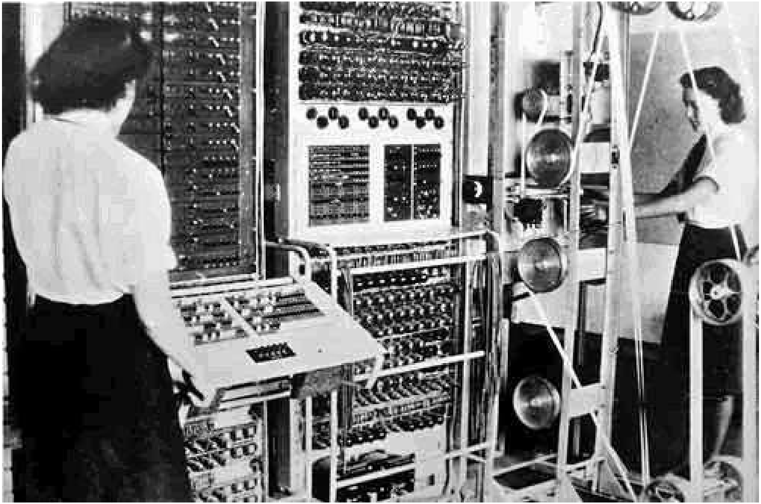
The History of the Digital Computer (1a)



The History of the Digital Computer (1b)



The History of the Digital Computer (1c)



The History of the Digital Computer (1d)

In electronics, a vacuum tube is a device used to amplify, switch, otherwise modify or create an electrical signal.

Vacuum tubes enabled the development of electronics technology (radio, TV, radar, HiFi, telephone networks, digital computers and industrial process control).

For most purposes, the vacuum tube has been replaced by solid-state semiconductor devices such as transistors and diodes. These are smaller, more efficient, more reliable and cheaper – either as discrete devices or integrated circuits.

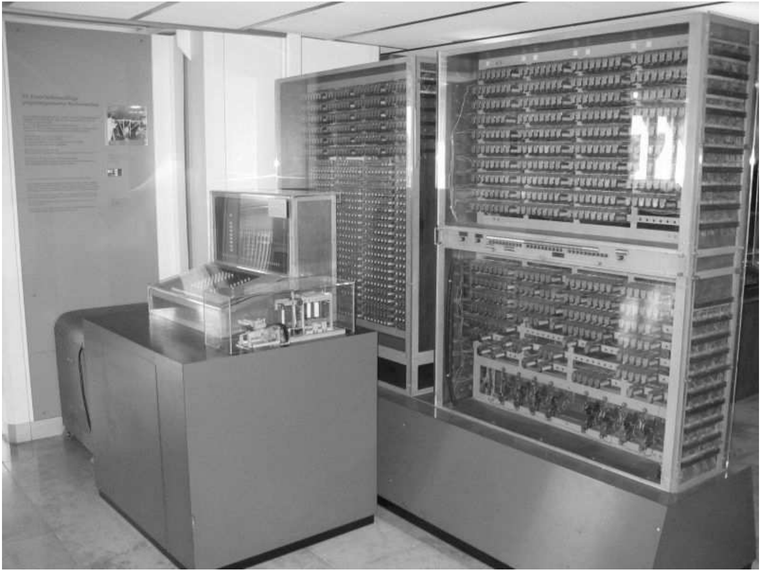
However, tubes are still in use, e. g. in monitors (cathode ray tube) and microwave ovens (magnetron).

The History of the Digital Computer (2)

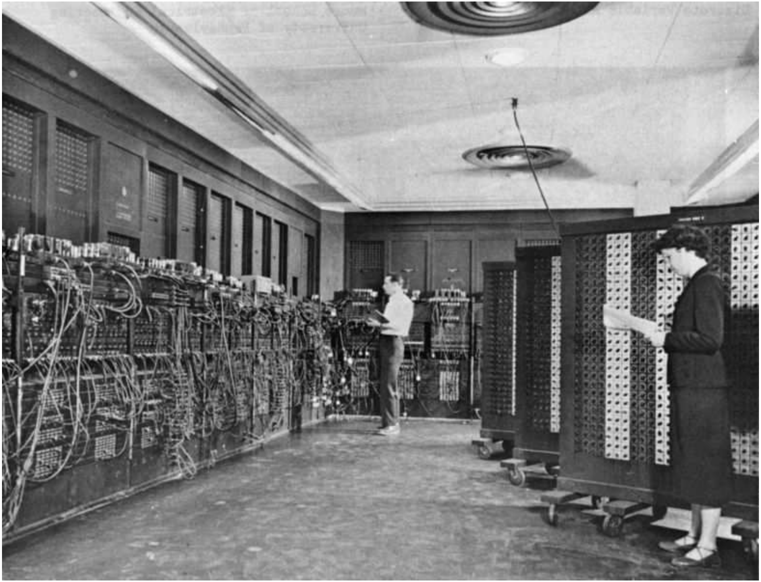
1941 Konrad Zuse builds the Z3 computer in Germany, the first programmable calculating machine with automatic control of its operations.

1946 ENIAC (Electronic Numerical Integrator and Computer), with 18,000 vacuum tubes, is dedicated at the U. of Pennsylvania. It was 8 by 100 feet and weighed 80 tons. It could do 5,000 additions and 360 multiplications per second.
174 kW; permanent operation
$+/-$ 0.2 ms, $\times$ 2.8 ms, $\div$ 24 ms, $\sqrt[2]{x} >$300 ms.

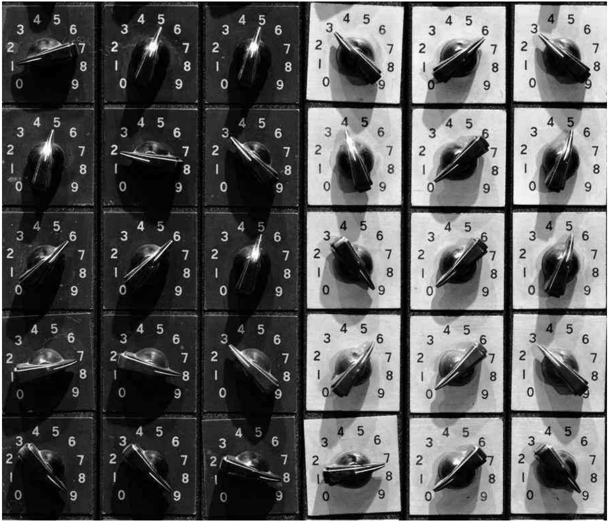
The History of the Digital Computer (2a)



The History of the Digital Computer (2b)



The History of the Digital Computer (2c)

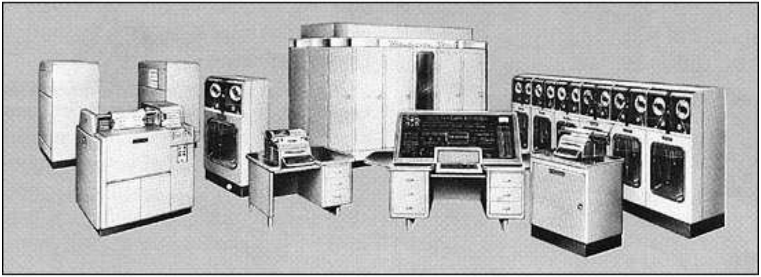


The History of the Digital Computer (3)

Computer	Year	Binary	Electronic	Programmable	Turing complete
Atanasoff-Berry	1939	Yes	Yes	No	No
Colossus	1941	Yes	Yes	By rewiring	No
Zuse Z3	1941	Yes	No	By punched film	Yes (1998)
ENIAC	1946	No	Yes	By rewiring	Yes
	1948	No	Yes	By resistor matrix	Yes

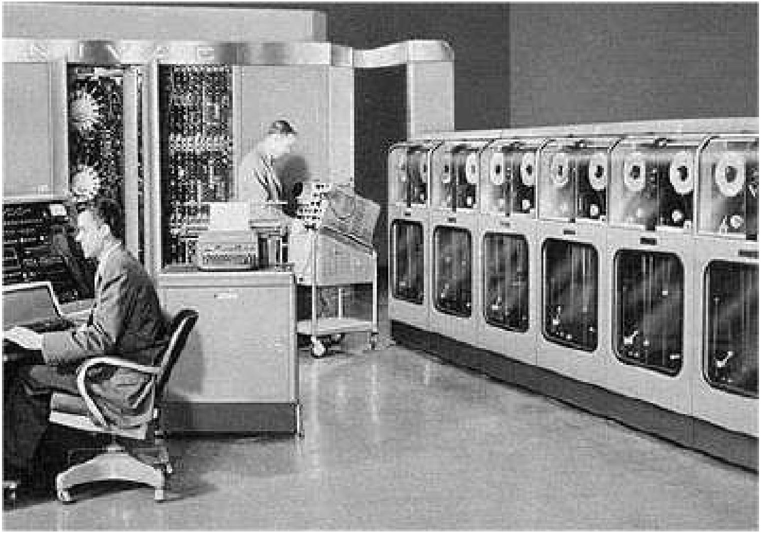
The History of the Digital Computer (4)

1951 Remington Rand builds the first commercial Universal Automatic Computer (UNIVAC).
5,200 vacuum tubes; 13 t; 125 kW; $>$35 m^2 of floor space
about 1,905 operations per second running on a 2.25 MHz clock.



Univac 1 computer, the first business data processor. Shown, left to right, High-speed printer with power supply and magnetic tape input unit, Unityper II, the central computer, the control console, the Universo 1 input-output unit, and the Uniprinter

The History of the Digital Computer (4b)



The History of the Digital Computer (4a)

The First NLP Application

Bi-lingual Dictionaries for Word-to-Word Machine Translation

1947 Donald Booth and D.H.V. Britten worked out a detailed code for realizing dictionary translation on a digital computer.

1948 R.H. Richens worked out a stem-affix encoding with a longest-match strategy for stem identification and translation.

Richen's Dictionary Encoding

An Example: Latin *amat* = love (3rd pers. sing.)

Dictionary Encoding:

stem dictionary: a-m
suffix dictionary: -at

Search Strategy:

for a given input (e.g. *amat*) find longest prefix (in this case *am*) that matches the stem dictionary;
match the remainder of the string (in this case *at*) against the stem dictionary.

Note: this is a direct precursor to later finite-state-encodings of computational lexica.

Definition of Computational Linguistics (1a)

Computational linguistics is the scientific study of language from a computational perspective.

Computational linguists are interested in providing computational models of various kinds of linguistic phenomena. These models may be “knowledge-based” (“hand-crafted”) or “data-driven” (“statistical” or “empirical”).

Definition of Computational Linguistics (1b)

Work in computational linguistics is in some cases motivated from a scientific perspective in that one is trying to provide a computational explanation for a particular linguistic or psycholinguistic phenomenon; and in other cases the motivation may be more purely technological in that one wants to provide a working component of a speech or natural language system.

http://www.aclweb.org/archive/misc/what.html

Definition of Computational Linguistics (2)

Computational linguistics is the application of linguistic theories and computational techniques to problems of natural language processing.

`http://www.ba.umist.ac.uk/public/departments/registrars/academicoffice/uga/lang.htm`

Definition of Computational Linguistics (3)

Computational linguistics is the science of language with particular attention given to the processing complexity constraints dictated by the human cognitive architecture. Like most sciences, computational linguistics also has engineering applications.

`http://www.cs.tcd.ie/courses/csll/CSLLcourse.html`

Definition of Computational Linguistics (4)

Computational linguistics is the study of computer systems for understanding and generating natural language.

```
Ralph Grishman, Computational
  Linguistics: An Introduction,
  Cambridge University Press 1986.
```

Central Goal of the Field

- build psychologically adequate models of human language processing capabilities on the basis of knowledge about the way in which humans acquire, store, and process language.
- build functionally correct models of human language processing capabilities on the basis of knowledge about the world and about language elicited from people and stored in the system.

Application Areas

- language processing by computers
  - machine translation
  - speech recognition
  - speech synthesis
  - man-machine interfaces
- language learning
- language description (computer-aided dialectology)
- language description (wordnets, grammar implementation)

Application Areas

- intelligent word processing: spelling correction, grammar correction
- document management
    - find relevant documents in collections
    - establish authorship of documents
    - catch plagiarism
    - extract information from documents
    - classify documents
    - summarize documents
    - summarize document collections

The Turing test for machine intelligence

From: Turing, A.M. (1950). Computing machinery and intelligence. Mind, 59, 433-460.

> *I propose to consider the question, "Can machines think?" This should begin with definitions of the meaning of the terms "machine" and "think."* ...
>
> *Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words.*

The Imitation Game

The new form of the problem can be described in terms of a game which we call the imitation game. *It is played with three people, a man (A), a woman (B), and an interrogator (C) who may be of either sex. The interrogator stays in a room apart from the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either "X is A and Y is B" or "X is B and Y is A."*

The Imitation Game (2)

The interrogator is allowed to put questions to A and B thus:

C: "Will X please tell me the length of his or her hair?"

Now suppose X is actually A, then A must answer. It is A's object in the game to try and cause C to make the wrong identification. His answer might therefore be:

"My hair is shingled, and the longest strands are about nine inches long."

The Imitation Game (3)

... The ideal arrangement is to have a teleprinter communicating between the two rooms.

... The object of the game for the third player (B) is to help the interrogator. The best strategy for her is probably to give truthful answers. She can add such things as "I am the woman, don't listen to him!" to her answers, but it will avail nothing as the man can make similar remarks.

The Imitation Game (4)

We now ask the question, "What will happen when a machine takes the part of A in this game?"

Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman? These questions replace our original, "Can machines think?"

Is Artificial Intelligence Possible?

- Can Machines Think?
- Can Machines Understand?
- Can Machines Learn?
- Can Machines Have Emotions?

Searle's Argument Against Strong AI

According to strong AI, the computer is not merely a tool in the study of the mind; rather the appropriately programmed computer really is a mind, in the sense that computers given the right programs can be literally said to understand and have other cognitive states. (Searle 1980, p. 417).

In strong AI, because the programmed computer has cognitive states, the programs are not mere tools that enable us to test psychological explanations; rather, the programs themselves are the explanations. (Searle 1980, p. 417)

Searle's Critique of Artificial Intelligence (2)

> *According to Strong AI, instantiating a formal program with the right input and output is a sufficient condition of, indeed is constitutive of, intentionality. As Newell (1979) puts it, the essence of the mental is the operation of a physical symbol system. (Searle 1980, p. 421)*

`Reference: Searle, J. R. (1980a), 'Minds, Brains, and Programs', Behavioral and Brain Sciences 3, pp. 417-42`

Searle's Chinese Room Argument

Suppose that I'm locked in a room and given a large batch of Chinese writing. Suppose furthermore (as is indeed the case) that I know no Chinese, either written or spoken, and that I'm not even confident that I could recognize Chinese writing as Chinese writing distinct from, say, Japanese writing or meaningless squiggles. To me, Chinese writing is just so many meaningless squiggles.

Searle's Chinese Room Argument (2)

Now suppose further that after this first batch of Chinese writing I am given a second batch of Chinese script together with a set of rules for correlating the second batch with the first batch. The rules are in English, and I understand these rules as well as any other native speaker of English. They enable me to correlate one set of formal symbols with another set of formal symbols, and all that 'formal' means here is that I can identify the symbols entirely by their shapes.

Searle's Chinese Room Argument (3)

Now suppose also that I am given a third batch of Chinese symbols together with some instructions, again in English, that enable me to correlate elements of this third batch with the first two batches, and these rules instruct me how to give back certain Chinese symbols with certain sorts of shapes in response to certain sorts of shapes given me in the third batch.

Searle's Chinese Room Argument (4)

Unknown to me, the people who are giving me all of these symbols call the first batch "a script", they call the second batch a "story" and they call the third batch "questions." Furthermore, they call the symbols I give them back in response to the third batch "answers to the questions." and the set of rules in English that they gave me, they call "the program."

Searle's Chinese Room Argument (5)

[...] it seems to me quite obvious in the example that I do not understand a word of the Chinese stories. I have inputs and outputs that are indistinguishable from those of the native Chinese speaker, and I can have any formal program you like, but I still understand nothing. [...]

A bit of Philosophy of Science

- **Theory:**

  A set of statements that determine the format and semantics of descriptions of phenomena in the purview of the theory

- **Methodology:**

  An effective theory comes with an explicit methodology for acquiring these descriptions

- **Application:**

  A theory associated with a methodology can be applied to tasks for which the methodology is appropriate.

Scientific Strategies

- **Method Oriented Approach:**

  devise or import a tool, a procedure or a formalism, apply it to a task and develop it further. Then (optionally) see whether it works for additional tasks

- **Task oriented Approach:**

  select a task; devise or import a method or several methods for its solution; integrate the methods as required to improve performance.