University of Malta
BSc(IT) HonsYear III

CSA305: Natural Language Processing

Lecture 2

Classic Examples

- History
- Machine Translation
- Turing Test
- ELIZA

Department of Computer Science and AI 2001/02
Lecturer: Michael Rosner

History of NLP

- 1950-60: Machine Translation
- 1960-70: Look Ma No Semantics
- 1970-80: AI takes the lead
- 1980-90: Grammar Formalisms
- 1990-00: Multilinguality, Multimodality

Machine Translation

At the end of the 1950s, researchers in the US, Russia, and Western Europe were confident that high quality machine translation of scientific and technical documents would be possible in a few years.

It was thought that all that was needed was a machine readable bilingual dictionary.

John	goes	to	work	by	car
Jean	va	au	travail	en	voiture

However, they underestimated the limitations of simple word-for-word translations.

Amusing Translations

- Les soldats sont dans le café

  The soldiers are in the coffee
- Je suis allé prendre le train

  I went capture the train
- La Cour de Justice envisage la creation d'un cinquième poste d'avocat général.

  The yard of justice envisages the creation of a fifth general avocado station.

Machine Translation is Hard

- Words with multiple meanings

  Ford: You should prepare yourself for the jump into hyperspace; it's unpleasantly like being drunk.
  Arthur: What's so unpleasant about being drunk?
  Ford: Just ask a glass of water.

- Sentences with multiple grammatical structures

  I saw the man on the hill with a telescope

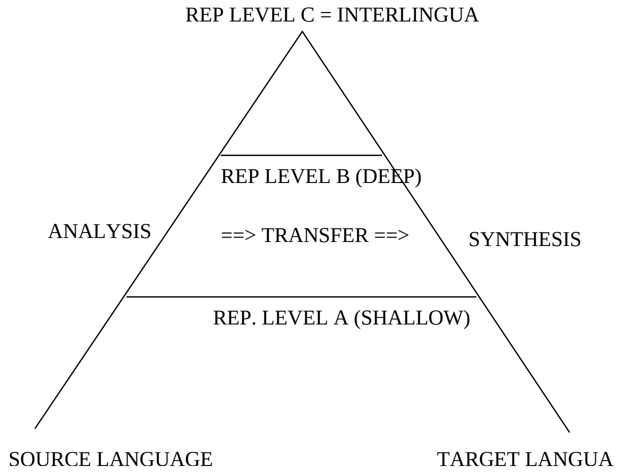
- Uncertainty about what a pronoun refers to

  There's a pile of inflammable trash next to your car. You're going to have to get rid of it.

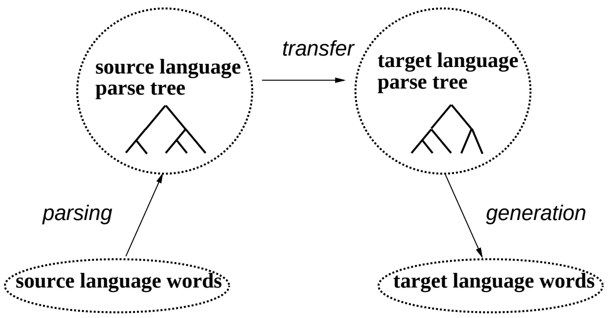
- The importance of context.

  I like you a lot

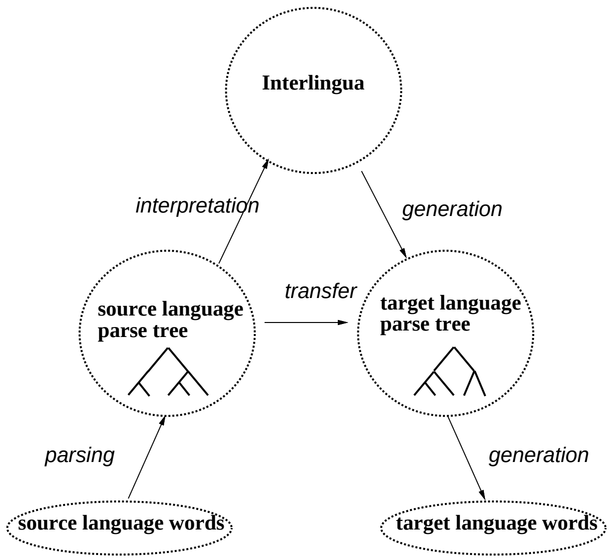
Architecture of MT Systems



Transfer Based MT Systems



Interlingua Based Systems



Turing Test and Chatterbots

- In October 1950, Alan Turing published a paper called Computing machinery and intelligence published in the Mind journal which was to change the course of computing. He chose to consider the question Can machines think? and a test which could show this.
- The test, known as The Turing Test was based on an old victorian parlour game. There is a man, a woman and an interrogator and the object of the game is for the interrorgator to ask questions to try to determine which is the man and which is the woman. The man can lie to any questions to try and confuse the interrogator.
- A Chatterbot is a program that attempts to simulate conversation, with the aim of at least temporarily fooling a real human being into thinking they were talking to another person.

ELIZA - Background

- In the beginning there was ELIZA ...
- Joseph Weizenbaum coded ELIZA at MIT during the years 1964-1966.
- The aim was to produce a very simple program, based on the presence of keywords in the input, capable of typed NL conversation.
- He developed a script for ELIZA which simulated a psychotherapist.
- Eliza encourages the patient to talk by reflecting the patient's statements back to him.
- The role played by the computer was very carefully chosen to disguise the program's lack of knowledge about language or about the world.

ELIZA - Example Interaction

Men are all alike.

IN WHAT WAY?

They're always bugging us about something or other

CAN YOU THINK OF A SPECIFIC EXAMPLE?

Well, my boyfriend made me come here

YOUR BOYFRIEND MADE YOU COME HERE?

He says I'm depressed much of the time

I'M SORRY TO HEAR YOU ARE DEPRESSED

Perhaps I could learn to get along with my mother.

TELL ME MORE ABOUT YOUR FAMILY

ELIZA - Problems Addressed

1. The identification of the most important keyword occurring in the input text.
2. The identification of some minimal context within which the chosen keyword appears.
3. The choice of an appropriate transformation rule and the making of the transformation itself.
4. The provision of a mechanism that permits ELIZA to respond intelligently when the input contains no keywords.
5. The provision of machinery to facilitate editing, particularly extension of the of the script.

ELIZA - How it Works

There is a database of keywords. each is associated with

- an integer indicating the keyword's priority
- a *decomposition rule* to match against the input

rule: (0	you	0	me)
input:(do	you	hate	me)
segments:1	2	3	4

  where 0 indicates a subphrase of arbitrary length.
- a *reassembly rule* n that constitutes an output specification, e.g

  (what makes you think that I 3 you?)

  where the number 3 indicates the third segment matched by the decomposition rule – in this case the word "hate"

Main Algorithm

Given a sentence S,

- find a keyword in S whose input pattern matches.
- If there is more than one, choose the one with the highest priority.
- Use the reassembly rule to generate the next sentence.
- If there are no matching keywords, generate an innocuous sentence.

ELIZA - Tricks

- Routine substitution of certain input words during scanning e.g.
  MY → YOUR; I→ YOU

```
I love my bicycle
(you love  0) => (WOULD YOU LIKE TO MARRY 3)
```

- On-the-fly composition of pre-canned responses

```
(MEMORY MY
 (0 YOUR 0 = LET'S DISCUSS FURTHER WHY YOUR 3)
 (0 YOUR 0 = EARLIER YOU SAID YOUR 3)
 etc
```

- Tagging of keywords according to class, e.g.

  `((MOTHER DLIST (/NOUN FAMILY))`

  to be referred to in decomposition rules

  `((0 YOUR 0 (/FAMILY) 0))`

Eliza - Discussion

1. It is in no sense a general solution to the problem of NL communication.
2. ELIZA is a remarkable example of *robust NLP*. It always produces an answer, no matter how ungrammatical or meaningless the input.
3. The program was written to demonstrate that given the right setting, people are inclined to ascribe intentions and thus, intelligence, to producers of utterances. The illusion is further fostered by a setting for which the program is not *required* to know anything.

J. Weizenbaum, *ELIZA - A computer Program for the Study of Natural Language Communication between Man and Machine*, CACM vol 9.1,pp36-43, 1966

Implementing ELIZA-like Systems

```
%% MAIN MAPPING FUNCTION %%

map(Input,Output) :-
 routinesubst(Old/New),
 subst(Input,Old/New,Output).

%% subst(IList,Substitution,OList)
%% returns Olist

subst([],_,[]).

subst([X|Irest],X/Y,[Y|Orest]) :-
 subst(Irest,X/Y,Orest).

subst([X|Irest],Z/Y,[X|Orest]) :-
 \+(X=Z),
 subst(Irest,X/Y,Orest).


%%%%%%  data %%%%%%%

routinesubst(i/you).
```


Enhancements

1. Read - Process - Write Loop.
2. Parallel substitution.
3. Keyword discovery.
4. Disassembly and reassembly rules.
5. Initiative handling.

SWI Prolog

Covington Chapter 2

Exercise 1

Implement 1 and 2 above.

Tutorial to discuss – PC Lab – Wednesday 11.00

Loebner Prize

- The Loebner Prize was set up by Hugh Loebner in 1991. A prize of $100,000 is available to the author of the first computer program to pass an unrestricted Turing Test.
- This test was proposed by Alan Turing in 1950, the principle of it being that if a computers responses were indistinguishable from those of a human, the computer could be said to be thinking. This prize has not yet been awarded.
- Every year there is an annual Loebner Prize contest, where there is a prize of $2000 avaiable for the most "human-like" of the entries that year. Many of the NLP systems we looked at for out presentation have been entered for the Loebner Prize.