Methods and tricks used in an attempt to pass the Turing Test

V.Bastin, D.Cordier
Department of Computer Science
The Flinders University of South Australia
Facultes Universitaires Notre Dame de la Paix (Namur - Belgium)

{vbastin,dcordier}@info.fundp.ac.be

Abstract

This paper describes differents methods and tricks in connection with our program which has been entered in the Loebner Prize competition that will happen on Sunday 11 January 1998, at the PowerHouse Museum in Sydney. Of course, this isn't exhaustive, there are other possible techniques but we aim to give the main ideas. We'll speak about the main modules of our program : Spelling correction, Different uses of WordNet, and Generation of comments. Our module used for spelling correction was developed on the basis of works by Brill [1], Brill and Marcus [2], Golding [3], Golding and Schabes [4], and Powers [5].

1. Introduction

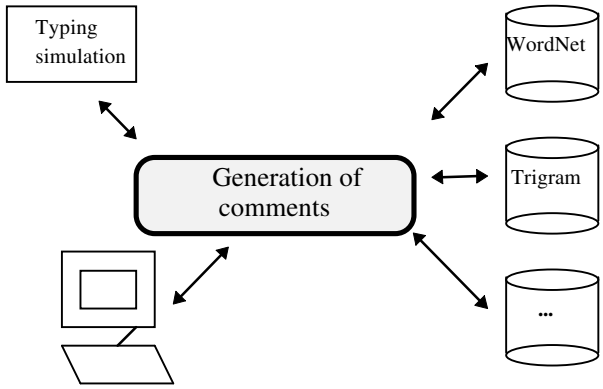
Alan Turing was a brilliant British mathematician who played an important role in the development of computers and developed a test that would serve as an indicator of intelligence for machines. A lot of researchers posed the Loebner Prize as the first formal instantiation of the Turing Test. To participate in this competition, we conceived a program that attempts to simulate the responses of a human being.
We'll begin to describe WordNet, which includes a classification of English words. Afterwards, we'll present the architecture of our system which we are programming at the moment. In this section, we'll briefly explain every module. Next, we'll give an example of interaction between our program and one human. In the same section, we'll show different processes of generating a response from the input of the user. Finally, we'll conclude by indicating our own position on this test, using knowledge that we have acquired during only two months of work in this area.

2. WordNet

WordNet is an on-line lexical reference system whose design is inspired by current psycholinguistics theories of human lexical memory. Actually, WordNet contains about 170,000 words, classified according to their part-of-speech (verbs, nouns, adjectives, adverbs). These sets are divided into semanticals categories (e.g. synonymous for nouns...). WordNet is completely described in the URL http://www.speech.cs.cmu.edu/comp.speech/Section1/Lexical/wordnet.html.

3. Architecture

To mimic some parts of human thought, we created different principal modules : Spelling Correction, Disambiguation between words, Generation of comments, Simulating human typing...



3.1. Spelling Correction

For spelling correction, we initially chose to create a prolog database, holding all words present in WordNet, indexed on every character, and reduce the sets of possible words at the time of typing. If we obtain an empty set, we can use a parallel process to search for every possible word, and await the end of typing to choose the most likely word using tri-grams. Another way is to build a database taking every word and the most common typing errors of this word. To determinate the most common errors, we can read training data from news, where many errors occur. Our last idea is to try to build a database containing every word from Wordnet. For every word, we modify

Veronique Bastin and Denis Cordier (1998) **Methods and tricks used in an attempt to pass the Turing Test.** In D.M.W. Powers (ed.) *NeMLaP3/CoNLL98 Workshop on Human Computer Conversation,* ACL, pp 275-277.

(change, delete, insert, transpose) one letter, and the new strings are added to the database. At the time of typing, if a word is not recognized then program would be able to find the most likely word in the context.

3.2. Disambiguation between words

Some words can occur in place of others, for many reasons. Powers [5] distinguishes six different types of reasons for substituted word errors. A good program would contain one module for every type of error.We haven't implemented the module of Disambiguation between words yet, but we'll try to do it for the Loebner Prize in January.

3.3. Generation of comments

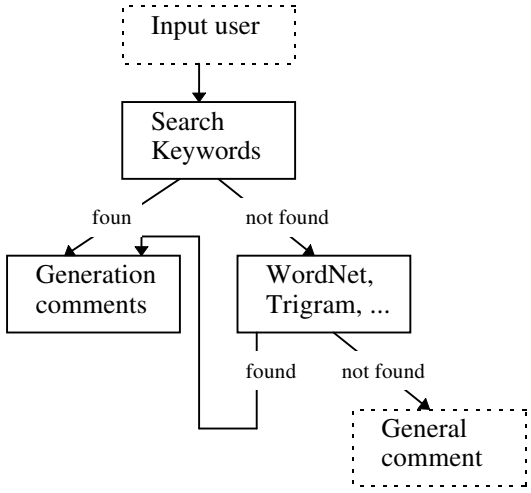
To try to generate comments in connection with current conversation, we analyse sentences typed by user, and try to find keywords. We use these keywords to determine an appropriate response. If we don't find keywords, we try to find hyponyms of the words of the sentence using WordNet. The best way would be to determine the part-of-speech and search for hyponyms of the principal parts. More over we would calculate distances between different hyponyms using Resnik's method[6] and choose the closest one, selecting a new keyword. To try to keep a line of conversation, we randomly ask the speaker some questions, about subjects present in our system. If after all processes, we don't find keywords, we just redisplay the input sentence, after having changed each pronom to its opposite (e.g. I went to town / You went to town?).

3.4. Simulating human typing

Now, we stop the process which displays the comment for a random number of milliseconds. It would be better to use a parallel process to display a first comment (e.g. I think...). At the same time, a separate process manages access to databases needed (WordNet, TriGram...) to display the main comments in a current conversation. This system allows us to simulate the human's speed of typing, but moreover we simulate human typing errors by using backspace and delete keys.

4. Demonstration

4.1. Flow chart



4.2. Example of interaction

Hello, I'm Elizabeth. How are you?
Fine thank you.

Good, I'm happy about that.
Does your family live in Australia?

Mmmm....
I have a small family. My family is in Europe.
Why are you in Australia?

I don't feel like speaking about that.
Have you a lot of friends?
Yes, I have friends.

Ok.
What's your religion?

I'm a Christian, but I don't like to speak about religion.
Are you going on holidays this year?

Mmm...
This year I 'm not going on holidays, because I 'm working in Australia. Generally, when I'm on holidays, I do nothing because I like sleep.

Where did you go on holidays last year?
Nowhere, I had no time.

No problem.
Would you like to come on holidays with me?

I'm so flattered.
Do you know Spanish?

I just know French but in this test, we must speak English.
Bye.
Bye.Nice to meet you.

5. Conclusion

To indicate our position in relation to the Loebner Prize, we think that this competition only answers a part of Turing's initial question, which is "Can machines think like a human?" and not "Can machines think?" (Turing).

We object to the fact that each judge has only a limited time to converse with each subject. This may not allow the judge to decide without hesitation if it's a machine or not. On the other hand for this competition, a cross-section of the community will be present. Our program needs luck to be able to converse with a speaker in a satisfying way. If a judge uses the pre-defined subjects in our system, the conversation will be coherent. However, we need to include a large number of pre-defined subjects. We also need to include systems of spelling correction which perform well, to try to retrieve keywords from words with spelling errors for instance. In fact we have only been working in this area for two months, we don't think that anyone could develop an intelligent machine so quickly. The future will tell. We can only wait and see...

6. References

Brill, E. A simple ruled-based part of speech tagger.

Brill, E, & Marcus, M.. Automatically acquiring phrase structure using distributional analysis.

Golding, A.R. (1995) A baysian hybride method for context-sensitive spelling correction *Mitsubishi Electric Information Technology Center America.*

Golding, A.R. & Schabes, Y. (1996). *Combining trigram-based and feature-based methods for context-sensitive spelling correction.*

Powers, D.M.W. (1997) *Learning and application of differential grammars.*

Resnik, P. (1995) *Disambiguating nouns groupings with respect to WordNet senses.*