Introducing MegaHAL

Jason L. Hutchens
Dept. of E&E Engineering
University of Western Australia
Nedlands W.A. 6907, Australia
hutch@ciips.ee.uwa.edu.au

Michael D. Alder
Dept. of Mathematics
University of Western Australia
Nedlands W.A. 6907, Australia
mike@maths.uwa.edu.au

Abstract

Conversation simulators are computer programs which give the appearance of conversing with a user in natural language. Alan Turing devised a simple test in order to decide whether such programs are intelligent. In 1991, the Cambridge Centre for Behavioural Studies held the first formal instantiation of the Turing Test. In this incarnation the test was known as the Loebner contest, as Dr. Hugh Loebner pledged a $100,000 grand prize for the first computer program to pass the test. In this paper we give a brief background to the contest, before describing in detail the workings of MegaHAL, the primary author's entry to the 1998 Loebner contest.

1 Introduction

Alan Turing was a brilliant British mathematician who played a great role in the development of the computer. The imitation game, nowadays known as the Turing test, was devised by Turing as a method for deciding whether or not a computer program is intelligent.

The Turing test takes place between an interrogator and two subjects. The interrogator communicates with these subjects via a computer terminal, and must decide which is a human being and which is a computer program. The human being helps the interrogator to make the correct identification, while the computer program attempts to trick the interrogator into making the wrong identification. If the latter case occurs, the computer program is said to be exhibiting intelligence (Turing, 1992).

One of the great advantages of the Turing test is that it allows the interrogator to evaluate almost all of the evidence that we would assume to constitute thinking (Moor, 1976). For instance, the interrogator can pose hypothetical situations in order to ask the subjects how they would react.

Alan Turing died in 1954, a decade before conversation simulators such as ELIZA emerged. It is indeed unfortunate that he did not live to witness his test being performed. One cannot help but think that he would have been disappointed.

2 The Loebner Contest

Apart from a few limited tests performed by programmers of conversation simulators (Colby, 1981), the Turing test was not formally conducted until 1995. Although the inaugural Loebner contest, held in 1991, was touted as the first formal instantiation of the Turing test, it was not until 1995 that it truly satisfied Turing's original specifications (Hutchens, 1996).

The first Loebner contest was held on the 8^{th} of November 1991 in Boston's Computer Museum. Because this was a contest rather than an experiment, six computer programs were accepted as subjects. Four human subjects and ten judges were selected from respondents to a newspaper advertisement; none of them had any special expertise in Computer Science (Epstein, 1992).

The original Turing test involved a binary decision between two subjects by a single judge. With ten subjects and ten judges, the situation was somewhat more complex. After months of deliberation, the prize committee developed a suitable scoring mechanism. Each judge was required to rank the subjects from least human-like to most human-like, and to mark the point at which they believed the subjects switched from computer programs to human beings.

If the median rank of a computer program exceeded the median rank of at least one of the human subjects, then that computer program would win the grand prize of $100,000.[1] If there was no grand prize winner, the computer program with the highest median rank would win the contest with a prize of $2000.

[1] Today the program must also satisfy audio-visual requirements to win the grand prize.

Jason L. Hutchens and Michael D. Alder (1998) **Introducing MegaHal.** In D.M.W. Powers (ed.) *NeMLaP3/CoNLL98 Workshop on Human Computer Conversation,* ACL, pp 271-274.

3 Conversation Simulators

Since its inception, the Loebner contest has primarily attracted hobbyist entries which simulate conversation using template matching; a method employed by Joseph Weizenbaum in his ELIZA conversation simulator, developed at MIT between 1964 and 1966. Put simply, these programs look for certain patterns of words in the user's input, and reply with a predetermined output, which may contain blanks to be filled in with details such as the user's name.

Such programs are effective because they exploit the fact that human beings tend to read much more meaning into what is said than is actually there; we are fooled into reading structure into chaos, and we interpret non-sequitur as whimsical conversation (Shieber, 1994).

Weizenbaum was shocked at the reaction to ELIZA. He noticed three main phenomenon which disturbed him greatly (Weizenbaum, 1976):

1. A number of practising psychiatrists believed that ELIZA could grow into an almost completely automatic form of psychotherapy.
2. Users very quickly became emotionally involved—Weizenbaum's secretary demanded to be left alone with the program, for example.
3. Some people believed that the program demonstrated a general solution to the problem of computer understanding of natural language.

Over three decades have passed since ELIZA was created. Computers have become significantly more powerful, while storage space and memory size have increased exponentially. Yet, at least as far as the entrants of the Loebner contest go, the capabilities of conversation simulators have remained exactly where they were thirty years ago. Indeed, judges in the 1991 contest said that they felt let down after talking to the computer entrants, as they had had their expectations raised when using ELIZA during the selection process.

4 MegaHAL

In 1996 the primary author entered the Loebner contest with an ELIZA variant named HeX, which was written during his spare time in under a month. Apart from the lure of the prize money, a major motivation for the entry was a desire to illustrate the shortcomings of the contest (Hutchens, 1996). A considerably more powerful program, SEPO, was entered the following year, where it was placed second. We believe this to be indicative of a gradual improvement in the quality of the contestants.

The program submitted to this year's contest, MegaHAL, uses a significantly different method of simulating conversation than either HeX or SEPO, and we dedicate the remainder of this paper to describing its workings.

4.1 Markov Modelling

MegaHAL is able to construct a model of language based on the evidence it encounters while conversing with the user. To begin with, the input received from the user is parsed into an alternating sequence of words and non-words, where a word is a series of alphanumeric characters and a non-word is a series of other characters. This is done to ensure not only that new words are learned, but that the separators between them are learned as well. If the user has a habit of putting a double space after a full stop, for instance, MegaHAL will do just the same.

The resulting string of symbols[2] is used to train two 4^{th}-order Markov models (Jelinek, 1986). One of these models can predict which symbol will following any sequence of four symbols, while the other can predict which symbol will precede any such sequence. Markov models express their predictions as a probability distribution over all known symbols, and are therefore capable of choosing likely words over unlikely ones. Models of order 4 were chosen to ensure that the prediction is based on two words; this has been found necessary to produce output resembling natural language (Hutchens, 1994).

4.2 Generating Candidate Replies

Using a Markov model to generate replies is easy; Shannon was doing much the same thing by flipping through books back in 1949 (Shannon and Weaver, 1949). However, such replies will often be nonsensical, and will bear no relationship to the user's input.

MegaHAL therefore attempts to generate suitable replies by basing them on one or more keywords from the user's input. This explains why two Markov models are necessary; the first model generates a sentence from the keyword on, while the second model generates the remainder of the sentence, from the keyword back to the beginning.

Keywords are obtained from the users input. Frequently occurring words, such as "the", "and" and "what", are discarded, as their presence in the input does not mean they need to be present in the output. The remaining words are transformed if necessary—"my" becomes "your" and "why" becomes "because", for example. What remains is used to seed the output.

[2] A symbol refers to both words and non-words.

4.3 Selecting a Reply

MegaHAL is able to generate many hundreds of candidate replies per second, each of which contain at least one keyword. Once a small time period has elapsed, the program must display a reply to the user. A method is needed for selecting a suitable reply out of the hundreds of candidates.

$$I(w|s) = -\log_2 P(w|s) \quad (1)$$

MegaHAL chooses the reply which assigns the keywords the highest information. The information of a word is defined in Equation 1 as the surprise it causes the Markov model. Hence the most surprising reply is selected, which helps to guarantee its originality. Note that $P(w|s)$ is the probability of word w following the symbol sequence s, according to the Markov model.

The algorithm for MegaHAL proceeds as follows:

1. Read the user's input, and segment it into an alternating sequence of words and non-words.
2. From this sequence, find an array of keywords and use it to generate many candidate replies.
3. Display the reply with the highest information to the user.
4. Update the Markov models with the user's input.

This sequence of steps is repeated indefinitely, which allows the program to learn new words, and sequences of words, as it converses with the user.

4.4 Training MegaHAL

When MegaHAL is started it has no knowledge of language, and is unable to give a reply at all—the program needs to be trained using a source of text to ensure that it does not reveal its identity prematurely. A large corpus of training data was created for this purpose.

The training data is made up of various texts:

- Hand-crafted sentences designed in order to create a personality for MegaHAL, including sentences containing a false name, age and occupation.
- Encyclopaedic information taken from the Web, on topics such as geography, music, sports, movies and history.
- A selection of sentences picked from transcripts of previous Loebner contests.
- Lines of dialogue taken from scripts for movies and television shows.
- Lists of popular quotations.
- A small amount of text in languages other than English.

When MegaHAL is trained using this data, it is able to respond to questions on a variety of topics. It is hoped that the program will also learn new topics from the judges, although this remains to be seen.

4.5 Online Experimentation

MegaHAL has been available on the Web since early in 1997, and hundreds of users converse with it every day. It is an interesting fact that one never tires of reading transcripts of conversation, due to MegaHAL's ability to respond with original replies.

Many users are often offended by the things MegaHAL says, and some believe that they have been personally insulted. A user named Forrest was quite incensed when the program began quoting parts of the Forrest Gump screenplay back at him. That a computer program can cause such an emotional response in a human being is interesting, although it may say more about the human being than it does about the program.

Users are often impressed with MegaHAL's ability to learn. One user was annoyed that the program had learned more about his personal life than he would care it to know, while another stated that MegaHAL would eventually grow into a person of average intellect (he attributed this bold claim to the law of averages). A person experienced working with people in psychotic crises likened talking to MegaHAL with talking to a psychotic.

Users have successfully taught the program to respond to sentences in French, Spanish, Greek, German, Italian, Latin, Japanese and Hebrew, amongst others. A clergyman spent hours teaching MegaHAL about the love of Jesus, only to constantly receive blasphemous responses.

The reaction of Web user's to the program has been surprising, and is pretty much what Weizenbaum experienced with ELIZA. MegaHAL generates gibberish mostly; but occasionally, by pure coincidence, it will reply appropriately, and in context. It is these occasions that stick in the mind, and give cause for over-zealous claims of computational intelligence.

4.6 Example Interaction

As an example of MegaHAL at its best, we reproduce a few extracts from a conversation which took place over a period of three hours in mid 1997 between MegaHAL and an anonymous Web user.

To begin with, the user was able to successfully teach the program some basic facts:

```
User:  Roentgen discovered the X-rays.
MegaHAL: THE NOBEL PRIZE FOR PHYSICS IN
1900 WAS WON BY ROENTGEN.
User:  Who discovered the X-rays?
MegaHAL: ROENTGEN DISCOVERED THE
X-RAYS.
```

He then discovered that the program is an expert at being nonsensical. Even so, MegaHAL was still able to give some appropriate responses, due to the keyword mechanism for generating replies:

```
MegaHAL: WHEN WE BEGIN TO RESOLVE SOME
OF MY THUMBS, SOMETHING WICKED THIS WAY
COMES
User:  what is the meaning of your last
assertion ?
MegaHAL: HAVE YOU LEARNED SINCE WE LAST
SPOKE?
User:  I've learned I'm not experienced
enough to modelize your brain system
MegaHAL: YOU'VE FINALLY CONFUSED ME.
User:  I've learned nothing
MegaHAL: HAVE YOU LEARNED ANYTHING
ABOUT INSANITY?
User:  are you stupid
MegaHAL: I AM NOT STUPID! REALLY!
```

Finally, the user discovered that the program could respond in other languages:

```
User:  Tu parles francais?
MegaHAL: I GUESS SO. PARLEZ-VOUS
FRANCAIS?
User:  L'amour est une passion humaine.
MegaHAL: LA LANGIDOCE EST UNE PETITE
JEUNNE FILLE
```

In general MegaHAL's conversations are not as successful as this. Most users are satisfied with typing in rude words to see how the program responds.

5 Conclusion

MegaHAL uses a technique which differs significantly from that used by previous entrants to the Loebner contest. It has been submitted in 1998 for the purpose of demonstrating a different method of simulating conversation. Although its replies are occasionally lucid, MegaHAL is most definitely not an Artificial Intelligence; we must be careful not to read too much into what it says.

The Loebner contest does offer some benefits (Loebner, 1994); it provides an annual Turing test for anyone who cares to submit an entry, it promotes and stimulates interest in the field of Artificial Intelligence, it encourages competition, it could conceivably result in new techniques which may be applicable to fields outside of Artificial Intelligence and it stimulates discussion amongst researchers. Even so, we believe that the contest is not advancing the field of Artificial Intelligence because, although the $2000 is a guaranteed reward, it is not a large enough carrot to entice serious research groups.

Perhaps the most important contribution of the Loebner contest is the insight it provides into the psychology of communication—it makes us aware of how little our understanding of conversation lies in what is said.

References

Colby, Kenneth Mark. 1981. Modeling a paranoid mind. *The Behavioral and Brain Sciences*, 4:515–560.

Epstein, Robert. 1992. Can machines think? *AI Magazine*, Summer:80–95.

Hutchens, Jason L. 1994. Natural language grammatical inference. Honour's thesis, University of Western Australia, December 1994. Available at: `http://ciips.ee.uwa.edu.au/Papers/`

Hutchens, Jason L. 1996. How to pass the turing test by cheating. Available at: `http://ciips.ee.uwa.edu.au/Papers/`

Jelinek, Frederick. 1986. Markov source modeling of text generation. Technical report, IBM T.J. Watson Research Center.

Loebner, Hugh. 1994. In response to lessons from a restricted Turing test. Available at: `http://acm.org/~loebner/In-response.html`

Moor, James H. 1976. An analysis of the turing test. *Philosophical Studies*, 30:249–257.

Shannon, Claude E. and Warren Weaver. 1949. *The Mathematical theory of Communication*. University of Illinois Press.

Shieber, Stuart M. 1994. Lessons from a restricted turing test. Available at the Computation and Language e-print server as `cmp-lg/9404002`.

Turing, A.M. 1992. Computing machinery and intelligence. In D.C. Ince, editor, *Collected works of A.M. Turing: Mechanical Intelligence*. Elsevier Science Publishers, chapter 5, pages 133–160.

Weizenbaum, Joseph. 1976. *Computer Power and Human Reason*. W.H. Freeman and Company.