A Survey of Chabot Systems through a Loebner Prize Competition

Luka Bradeško*, Dunja Mladenić*

* Artificial Intelligence laboratory, Jozef Stefan Institute,
Ljubljana Slovenia
{luka.bradesko, dunja.mladenic}@ijs.si

Abstract

Starting in 1966 with the introduction of the ELIZA chatbot, a great deal of effort has been devoted towards the goal of developing a chatbot system that would be able to pass the Turing Test. These efforts have resulted in the creation of a variety of technologies and have taken a variety of approaches. In this paper we compare and discuss the different technologies used in the chatbots which have won the Loebner Prize Competition, the first formal instantiation of the Turing Test. Although there was no game changing breakthrough in the chatbot technologies, it is obvious they evolved from the very simple pattern matching systems towards complicated patterns combined with ontologies and knowledge bases enabling computer reasoning.

Pregled najboljših programov za klepetanje z računalnikom iz Loebnerjevega tekomovanja

Vse od leta 1966 naprej, ko se je pojavil prvi program za klepet ELIZA, se poskuša razviti program ki bi bil sposoben opraviti Turingov test. V tem času je bilo razvitih veliko različnih rešitev in pristopov k reševanju tega kompleksnega problema. V tem članku primerjamo in opišemo pristope pri programih za klepetanje ki so zmagali na Loebnerjevem tekmovanju. Loebnerjevo tekmovanje je prva formalna izvedba Turingovega testa in se izvaja že vse od leta 1991. Kljub temu, da do danes še nobenemu programu ni uspelo prestati testa, so močno napredovali. Iz zelo enostavnih algoritmov z iskanjem vzorcev besedila, so se z leti razvili v kompleksne sisteme prepoznavanja vzorcev, ki vključujejo tudi ontologije in tako do neke mere že omogočajo tudi računalniško sklepanje.

1. Introduction

There are numerous approaches to human-computer interaction. One of them is via natural language (NL), which again has more sub-approaches and goals. In this paper we focus on chatbots, which are gaining popularity again due to success of virtual assistants such as Siri, Evi, S-Voice, Jeannie, CallMom and others.

The main purpose and idea of the so called chat-bots is that the computer is performing a natural language conversation with human clients which should be as human-like as possible. Based on the task bot was made for, the conversations then usually serves some specific purpose such as searching the web, organizing files on the computer, setting up appointments, etc.

Currently the biggest challenge that existing chat-bots have is maintaining of the context and understanding the human inputs and its responses. Most of the existing bots still work just on the pattern matching of inputs and then trying to find a scripted response which matches the input. This approach however cannot result in a fully satisfying conversation or lead a conversation with some specific purpose.

Due to the obvious drawbacks of scripted responses, developers and researchers kept adding new functionalities to the existing ways how chatbots works, converging mostly to the use some sort of ontologies and remembering facts from the conversation. While these improvements made chatbots much more successful, at the same time introduced a number of different approaches, systems and solutions to the same problem.

The goal of this paper is to make a survey of chatbot technologies and approaches and thus make it easier for a developer and/or a researcher on to which technology to use for the research or further development of the chatbot system.

2. Early chatbots

There were numerous chatbots and chatbot technologies already before the first Loebner competition, mostly in games and focused domain expert systems. It is not known how well they performed and they were never compared against each other.

The very first known chatbot was Eliza, which was developed in 1966. Its goal was to behave as a Rogerian psychologist. It used simple pattern matching and mostly returned users sentences in a form of questions. Its conversational ability was not very good, but it was enough to confuse people at a time when they were not used to interact with computers and to start the development of other chatbot systems. The very first online implementation of Eliza was done by the researches at Jozef Stefan Institute in Ljubljana, Slovenia and is still available[1] for testing.

The first such a system that was actually evaluated using some sort of Turing Test was PARRY (Colby, 1975). Parry was designed to talk as a paranoid person. Its transcripts were given to psychiatrists together with transcripts from real paranoia patients for comparison. The psychiatrists were able to make the correct identification only 48% of the time.

3. Loebner Prize Competition

Loebner Prize Competition[2] (Loebner prize for artificial intelligence) is an annual competition for conversational agents (chatbots), where they are being

[1] http://www-ai.ijs.si/eliza-cgi-bin/eliza_script
[2] http://www.loebner.net

Year	Chatbot	Technology	Language Tricks
1991	PC Therapist III (Weintraub, 1986)*	parsing, pattern matching, word vocabulary, remembers sentences	non sequitur, canned responses
1992	PC Professor (Weintraub, 1986)*		
1993	PC Politician (Weintraub, 1986)*		
1994	TIPS (Whalen, 1994; Hutchens, 1997)	Pattern matching, database like system	Model of personal history
1995	PC Therapist (Weintraub, 1986)*	Same as in 1991	Same as in 1991
1996	HeX (Hutchens, 1997)	Pattern matching, Markov chain models to construct some replies	database of trick sentences, Model of personal history, not repeating itself
1997	CONVERSE (Levy, 1997)	Statistical parser, pattern matching, modular with weighted modules, WordNet synonyms, list of proper names, ontology, database for storing facts	Proactivity
1998	Albert One (Garner, 1995)	Pattern matching, hierarchical composition of other chat bots (Eliza, Fred, Sextalk)	Proactive monologues
1999	Albert One (Garner, 1995)		
2000	A.L.I.C.E (Wallace, 2003)	AIML (Artificial Intelligence Mark-up Language – advanced pattern matching)	
2001	A.L.I.C.E (Wallace, 2003)		
2002	Ella (Copple, 2009)	Pattern matching, phrase normalization, abbreviation expansion, WordNet	Monologues, not repeating itself, identify gibberish, play knock-knock jokes
2003	Jabberwock (Pirner, 2003)	Parser, pattern matching – simpler than AIML, Markov Chains, Context free grammar (CFG)	
2004	A.L.I.C.E (Wallace, 2003)	Same as in 2000	
2005	George (Carpenter, 2006)	Based on Jabberwacky chatbot. No pattern matching or scripts. Huge database of people's responses.	
2006	Joan (Carpenter, 2006)		
2007	UlrtraHAL by Robert Medeksza*	Combination of VB code and pattern matching scripts	
2008	Elbot (Roberts, 2007)*	Commercial NLI system	
2009	Do-Much-More (Levy, 2009)*	Commercial property of Intelligent Toys Ltd.	
2010	Suzette (Wilcox, 2011)	ChatScript (AIML successor. Concepts, triples, variables)	
2011	Rosette (Wilcox, 2011)		
2012	Chip Vivant (Embar, 2011)	Not publicly disclosed. Common ontology and AI, responses taken from ChatScript (but not in ChatScript format or engine).	

Table 1: List of Loebner winners with appropriate technologies

tested via a method called Turing Test (Turing 1950). Loebner Competition is known to be the first formal instantiation of a Turing Test.

There is a controversy whether this competition is really contributing to the development of AI, or it is blocking it (Shieber, 2006; Maughan, 2002; Hutchens, 1997; French, 1990). The doubt is because the competition is forcing chatbots to pretend to be a human which causes bots to simply pretend they are thinking without real intelligence. Some of the chatbots even fake the spelling mistakes and corrections. Another stated flaw is that the competition causes people to work apart instead to collaborate and thus lead to many incompatible chatbot technologies.

However this competition still methodologically compares chatbot technologies, rates them in a conversational sense and thus gives some sort of a general feedback over the used technologies.

3.1. Turing Test

Turing Test is also known as the Imitation Game. In this test, the goal for the chatbot is to maintain a conversation which is indistinguishable from a human conversation. The usual way to apply the test is that there is a human observer (judge), who is asking questions or having a conversation with someone over the computer link. That someone can be a computer (chatbot) or a person. If on the other side there is a chatbot and the judge would think it is a person, then the chatbot would pass the test.

3.2. Winning chatbots

Regardless of the fact that none of the existing chatbots were able to pass the Turing Test, each there is a winner of the Loebner Prize Competition that appears most human from all the competing chatbots. The list of each year's winners together with the used technologies can be seen on the Table 1. We tried to separate the technology part into the technical approaches and algorithms and the language and approach tricks used to confuse judges on the Turing Test. The technologies are explained in the following chapters in more detail. The winners marked with asterisk (*) are commercial programs and thus their technologies and internal structure is not publicly available.

4. Technical approaches and algorithms

4.1. Pattern Matching

This is by far the most common approach and technique used in chatbots. Variations of some pattern matching algorithm exist in every existing chatbot system.

The pattern matching approaches can vary in their complexity, but the basic idea is the same. The simplest patterns were used in earlier chatbots such as ELIZA and PC Therapist. For example:

Pattern: “I need a ?X”

Response: “What would it mean to you if you got a ?X?”

4.2. Parsing

Textual Parsing is a method which takes the original text and converts it into a set of words (lexical parsing) with features, mostly to determine its grammatical structure. On top of that, the lexical structure can be then checked if it forms allowable expression (syntactical parsing).

The earlier parsers were very simple, looking for recognizable keywords in allowed order. Example of such parsing would be that sentences “please take the gold” and “can you get the gold” would be both parsed into “take gold”. With this approach the chatbot with a limited set of patterns can cover multiple input sentences.

The more complicated parsers used in latter chatbots do the complete grammatical parsing of the natural language sentences.

4.3. Markov Chain Models

The Idea behind Markov Chain Models is that each occurrence of a letter or a word in some textual dataset occurs with a fixed probability. The order of a model means how many consecutive occurrences the model takes into the account. For example if an input text is “agggcagcgggcg”, then the Markov model of order 0 predicts that letter ‘a’ occurs with a probability 2/13. The model with order 1 would state that each letter still occurs with a fixed probability, but that probability depends on the letter before.

In chatbots the Markov Chain Models were being used to construct responses which are probabilistically more viable and thus more correct. In some cases (HeX) these models were even used to generate a nonsense sentence that sounds right, as a failback method.

4.4. Ontologies (semantic nets)

Ontology or semantic network as it is called in some chatbot systems is a set of hierarchically and relationally interconnected concepts. These concepts can have natural language names and can be used directly in chatbots, to figure out hyponyms, synonyms and other relations between the concepts. Example of such an ontology which is often used or at least tried to be used in chatbots is OpenCyc[3] (Lenat, 1995). The advantage of the ontologies is that the concepts are interconnected into a graph, which enables computers to search through and using special reasoning rules even imply new statements (reasoning).

4.5. AIML

AIML’s syntax is XML based and consists mostly of input rules (categories) with appropriate output. The pattern must cover the entire input and is case insensitive. It is possible to use a wildcard (*) which binds to one or more words. The simplest example of it can be written like seen on Figure 1. Due to simple and effective explanation, this and as well the other examples were taken from the paper Beyond Façade: Pattern Matching for Natural Language Applications (Wilcox, 2011).

```
<Category>
<pattern> I NEED HELP * </pattern>
<template>Can you ask for help in the form of a question?
</template>
</category>
```

Figure 1: Simple AIML rule (pattern).

The real power of AIML lies in its ability to recursively call itself (Wallace, 2003; Wilcox, 2011). It can submit input to itself using the *<srai>* tag and the contents of * using *<star/>*. Example of such recursion can be seen on Figure 2, where the AIML engine forwards everything before the phrase “right now” to another pattern. The second pattern then forwards everything after the phrase “can you please”.

```
=> Can you please tell me what LINUX is right now?
<category>
<pattern> * RIGHT NOW <\pattern>
<template> <srai><star/></srai></template>
</category>
=> CAN YOU PLEASE TELL ME WHAT LINUX IS
<category>
<pattern> CAN YOU PLEASE * <\pattern>
<template> <srai> Please <star/></srai></template>
</category>
```

Figure 2: AIML recursion.

AIML allows chatbots to have topics which give it a way to prioritize the patterns. It has the *<that>* pattern as well, which if it matches the output of the previous sentence it has priority over the other rules.

4.6. ChatScript

ChatScript is successor of the AIML language. It focuses on the better syntax which makes it easier to maintain. It fixes the zero word matching problems and introduces a bunch of additional functionalities such as concepts, continuations, logical and/or, variables, fact triples and functions. With these functionalities it tries to cover the need for ontologies inside the scrip itself. Example of a script defining a concept of meat and one pattern can be seen on Figure 3.

```
concept: ~meat ( bacon ham beef meat flesh veal lamb
chicken pork steak cow pig )
s: ( I love ~meat ) Do you really? I am a vegan.
```

Figure 3: Chatbot concept definition and simple pattern.

---

[3] http://www.opencyc.org/

5. Language approaches and tricks

5.1. Non Sequitur

Non sequitur (Latin) is an argument that has conclusions which does not imply from its premises. Example from everyday speech would be: “Life is life and fun is fun, but it’s all so quiet when the goldfish die.”

5.2. Simulating keystrokes and typing errors

The chat protocol that is used in Loebner Competitions works in a way that the judges see the sentences as they are being typed. This forces the chatbots to “pretend” they are tying word by word. Some of the bots even fake the spelling mistakes and backspacing.

5.3. Canned responses

Canned responses are predefined (hard coded) responses to questions. To some extent all of the chatbots patterns could be counted as canned responses if bot only uses these. This would vastly increase the number of patterns and would make them even more unmanageable, so these responses are usually used only for things which cannot be covered with the main chatbot technology.

5.4. Model of personal history

With the goal for a chatbot to appear more convincing, developers are inserting a personal story (imaginary or based on a real person) into chatbot responses. This includes memories from the past, childhood stories, parents, interests, political and religious views etc.

6. Conclusions

Through the years of Loebner Prize Competitions we can see how the chat technologies evolved from the very simple pattern matching systems, over the statistical models of chats, towards complicated patterns in combination with ontologies and knowledge bases. It can be argued that even the newest approaches (ChatScript, AIML) are still just a small improvement over the ELIZA pattern matching idea and that the biggest improvement is the amount of scripts written for it. We agree that there is some truth in it; however it is notable that the recent developments, especially with ChatScript the chatbots are moving out of the scripted era.

It is obvious that there is a trend towards semantics, which can lead to a conclusion that future chat bots will evolve from the pattern matching, towards more semantic approaches and will probably start to incorporate more and more computer reasoning systems.

Independently of Loebner competitions and other chat bot systems, IBM in 2004 started developing a question answering system (Watson), which won the show in 2011. Technically it is not a chatbot, since it is only able to answer questions, but their research currently leads into that direction as well. On top of more than 100 different text processing approaches, they used ontologies such as DBPedia, WordNet, and Yago for support to other techniques and to enable reasoning, which goes in hand with other newer chatbot approaches.

7. References

Colby K. M., 1975, Artificial Paranoia: A computer program for the study of natural language communication between man and machine, Communications of the ACM, vol. 9, pp. 36-45.

Copple K. L., 2009. Bringing AI to Life: Putting Today’s Tools and Re-sources to Work.

Embar M., 2011. Chip Vivant, http://www.chipvivant.com/motivations-and-functionality/

French R., 1990. Sub cognition and the limits of the Turing test, Mind, vol. 99, pp. 53-65.

Fryer L., Carpenter R., 2006. Emerging technologies: Bots as language learning tools. Language Learning and Technology, 10(3). Pp. 8-14, available at http://llt.msu.edu/vol10num3/pdf/emerging.pdf

Garner R., 2005. Multifaceted Conversational Systems, Colloquium on Conversational Systems, University of Surrey, available at http://www.robitron.com/Robby/Multifaceted.ppt

Hutchens L. J., 1997. How to pass the Turing Test by Cheating, University of Western Australia, available at http://www.nyu.edu/gsas/dept/philo/courses/mindsandmachines/Papers/hutchens96how.pdf

Lenat, D. B., 1995. Cyc: A Large-Scale Investment in Knowledge Infrastructure. Communications of the ACM, 38(22).

Levy D., Catizone R., Battacharia B., Krotov A., Wilks Y., 1997. CONVERSE: A Conversational Companion, In Proceedings of 1st International Workshop on Human-Computer Conversation, available at http://staffwww.dcs.shef.ac.uk/people/Y.Wilks/papers/converse.pdf

Levy D., 2009. DO-MUCH-MORE Chatbot Wins 2009 Loebner Prize for Computer Conversation, News from the Benelux Association for Artificial Intelligence, Vol. 26, No. 4, p. 67-78, available at http://www.unimaas.nl/bnvki/archive/2009/26.4.pdf

Maughan R., 2002. Conversational Agents, School of Computer Science and Information Technology, University of Nottingham

Pirner J., 2003. About Jabberwock, available at http://www.abenteuermedien.de/jabberwock

Roberts F., Gülsdorff B., 2007. Techniques of Dialogue Simulation, Lecture Notes in Computer Science p. 420-421.

Shieber S. M., 2006, Does the Turing Test Demonstrate Intelligence or Not?, AAAI-06, Boston, Ma.

Turing A.M., 1950. Computing Machinery and Intelligence, MIND the Journal of the Mind Association, vol. LIX, pp. 433-460.

Wallace R., 2003. The elements of AIML style. ALICE AI Foundation, available at http://www.alicebot.org/style.pdf

Weintraub J., 1986, History of the PC Therapist, available at http://www.loebner.net/Prizef/weintraub-bio.html.

Whalen T., 1994, My experience at Loebner prize, available at http://thomwhalen.com/ThomLoebner1994.html

Wilcox, B., 2011. Beyond Façade: Pattern Matching for Natural Language Applications, available at http://www.gamasutra.com/view/feature/134675/beyond_fa%C3%A7ade_pattern_matching_.php