An Educational Talking Toy based on an Enhanced Chatbot

Eberhard Grötsch[1], Alfredo Pina[2], Roman Balandin[1], Andre Barthel[1] and Maximilian Hegwein[1]
[1]*Fakultät Informatik und Wirtschaftsinformatik, Hochschule für Angewandte Wissenschaften,*
Sanderheinrichsleitenweg 20, Würzburg, Germany
[2]*Departamento de Informática, Universidad Pública de Navarra, Campus Arrosadia, Pamplona, Spain*

Keywords: Chatbot, Autonomous Toy, Children, Memory, Educational Patterns, Educational Talk.

Abstract: Children are often motivated in their communication behaviour by pets or toys. Our aim is to investigate, how communication with “intelligent“ systems affects interaction of children. Enhanced chatbot technology – hidden in toys - is used to talk to children. In the Háblame-project (started as part of the EU-funded Gaviota project) a first prototype talking German is available. We outline the technical solution, and discuss further steps.

1 INTRODUCTION

Within the Gaviota project (Bossavit et al., 2014), we worked on a system to investigate how oral communication with “intelligent“ systems affects the oral interaction of children with typical or untypical development.

To answer that question, we started to develop those intelligent systems first. While talking and understanding systems are widely in use (e.g. Siri, provided on some Apple devices), we did not find a system, which can be configured to our special needs.

So we decided to develop our own system.

2 REQUIREMENTS

Our goal is to develop a multi-client educational system, which talks to different children having individual knowledge about them (e.g. homework, friends, parents, favourite games). We assume that it also can talk to people suffering from dementia about their personal daily routine. The system should talk like an adult human.

A special tool has to be provided to enable non-programmers to administrate basic knowledge about individual persons. Client data have to be protected against unauthorized access.

Different from standard chatbots, the system does not just answer questions of the clients, but it is able to start a conversation, or able to begin with a new topic.

3 LEVELS OF TALKING

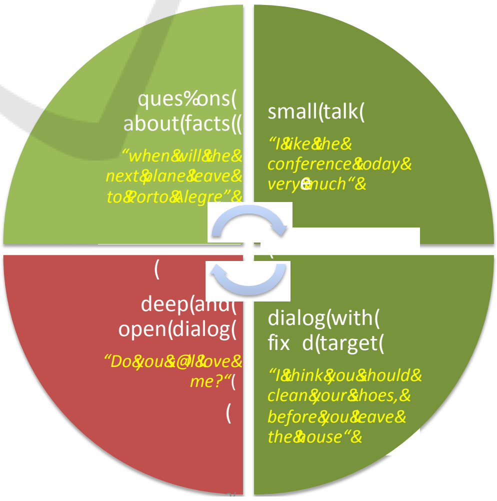

Figure 1: Levels of talking (Bossavit et al., 2014)

There are different levels of talking (fig. 1): from just checking facts to deep and open dialogs. Only answering questions about facts does not meet the requirements, so small talk should also be covered in the project. Usually chatbots are used to cover small talk, but that is not sufficient for educational purposes. The purpose of the system requires meaningful and target oriented talk.

Our system does not try to cover deep and open dialogs.

So the first attempt to solve the problem given is to use somehow enhanced chatbot technology.

Groetsch E., Pina A., Balandin R., Barthel A. and Hegwein M..
An Educational Talking Toy based on an Enhanced Chatbot.
DOI: 10.5220/0005452403600363
In *Proceedings of the 7th International Conference on Computer Supported Education* (CSEDU-2015), pages 360-363
ISBN: 978-989-758-107-6

4 PREVIOUS WORK

4.1 The Beginning: Eliza

Already in 1966 Weizenbaum (Weizenbaum, 1966) implemented Eliza, a talking system (based on text strings, without speech). The Eliza implementation used to react to a limited number of key words (family, mother, ...) to continue a dialog. Eliza had no (deep) knowledge about domains - not even shallow reasoning, rather a smart substitution for strings. Modern versions of Eliza can be tested on several websites, e.g. (ELIZA, 2013).

4.2 Traditional Dialog Systems

Most dialog systems (e.g. the original Deutsche Bahn system giving information about train time tables, or the extended system by Philips) are able to guide people who call a hotline and execute standardized business processes (delivering account data, changing address data, etc.). The dialogs in such systems are predefined and follow strict rules. They work well, but within a limited domain.

Chatbots are more flexible. Depending on the size of their database, they can talk without fixed sequences about a wide area of topics (cf. 4.4).

4.3 Natural Language Processing (NLP)

A spectacular demonstration of natural language processing was given by IBM's artificial intelligence computer system Watson in 2011, when it competed on the quiz show Jeopardy! former human winners of that popular US television show (JEOPARDY, 2011).

IBM used the Apache UIMA framework, a standard widely used in artificial intelligence (UIMA, 2013). UIMA means "Unstructured Information Management Architecture".

The source code for a reference implementation of this framework is available on the website of the Apache Software Foundation.

Systems that are used in medical environments to analyse clinical notes serve as examples.

4.4 Chatbots

Chatbots like Siri or Alice are popular among users of mobile phones to ease interaction with the system. Their domains are small talk, access to apps like calendars, or access to services like weather or traffic information.

Today chatbot technology is accepted in those areas mentioned, but it is not widely used.

However, in general chatbots do not have special knowledge about their users, and they do not initiate interaction with the user.

This is a severe limitation to educational systems – the system has to have at least basic knowledge about its clients, and it has to have a concept how to start communication and how to overcome breaks in the interaction. Therefore we concluded, that chatbots are a suitable tool to begin with, but we had to enhance them to meet the additional requirements of educational systems.

5 THE "HÁBLAME" PROJECT

5.1 Concept of a Dialog System

In the beginning, we intended to achieve our goals by real natural language processing (NLP), i.e., we studied basic concepts of NLP, and started with syntax and semantic parsing (Grötsch et.al., 2013).

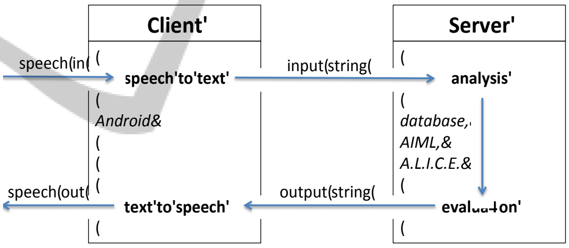

Figure 2: Concept of a dialog system based on (Schmitt, 2014).

But we soon recognized, that it is rather tedious to build such an NLP system based on deep language understanding – although there are some powerful tools available.

So we decided to use chatbot technology as described above. Fig. 2 shows the basic overall architecture of our system: we use a client (smartphone) for speech-to-text and text-to-speech conversion, the input strings recognized are sent to a server, which implements the enhanced chatbot functions and generates an output string, which is returned to the client.

5.2 Chatbot Architecture

None of the chatbots available fulfill the requirements of section 2. Therefore we looked for an open system, which we could adapt to our needs.

We selected the "Artificial Linguistic Internet Computer Entity" (A.L.I.C.E.)

"A.L.I.C.E. is an artificial intelligence natural language chat robot based on an experiment specified by Alan M. Turing in 1950. The A.L.I.C.E. software utilizes AIML, an XML language we designed for creating stimulus-response chat robots." (Wallace, 2014). It is released by the Alice A.I. Foundation under the GNU General Public License.

So A.L.I.C.E. uses the *Artificial Intelligence Markup Language (AIML)* to store information. It can generate new phrases, but they are stored under supervision of a human to prevent wrong input.

5.3 Client Functions

The clients are Android based systems. Therefore the built-in speech to text system of Android is used, which produces text strings. Those strings can be forwarded to an app which also handles the communication to and from the server.

Text to speech is covered by an app provided by Google.

5.4 Server Functions

The server runs A.L.I.C.E. and a file which contains AIML, which describes the patterns and templates of the dialog. (Our work is based on AIML 2.0, and its reference implementation ProgramAB)

A.L.I.C.E. is enhanced by the individual personal memory (IPM, cf. 5.5). To act as web server it is based on Apache Tomcat.

A short example shows the structure of AIML (table 1).

Each category (fundamental unit of knowledge in AIML) consists at least of a pattern (human input) and a template (answer of the bot). Wildcards ("*") grant more flexibility in describing the input. In table 1, line 2 the wildcard "*" covers any arbitrary input, and whatever the human says in the beginning, the answer of the bot will be: "What is your name?" The content of the wildcard in the pattern of line 6 is used in the answer of template line 8. Line 14 and line 20 are references to a already defined template: to the template of line 6, and the answer to all three patterns (line 6, 14 and 20) will be template line 8: "* is a beautiful name".

AIML offers more elements than the ones shown: e.g. several prioritized wildcards, topics, which cover several categories, or *random*, which selects by accident, or conditional branching.

Table 1: Example of simple AIML (Balandin, Hegwein, Barthel, 2014)

```
1 <category>
2    <pattern>*</pattern>
3    <template> What is your name? </template>
4 </category>

5 <category>
6    <pattern>my name is * </pattern>
7    <template>
8        <star index="1" /> is a beautiful name.
9    </template>
10 </category>

11 <category>
12   <pattern>i am called * </pattern>
13   <template>
14      <srai>my name is <star/></srai>
15   </template>
16 </category>

17 <category>
18   <pattern>one calls me * </pattern>
19   <template>
20      <srai>my name is <star/></srai>
21   </template>
22 </category>
```

5.5 Enhancing AIML

It is necessary to implement four new features to add the functionality of the project requirements:

1. data structures to store individual data of users (IPM – individual personal memory), (Balandin et.al., 2014)
2. multi-client properties,
3. timeout functions to recover from breaks / start a dialog, (Balandin et.al., 2014),
4. an editor tool for adding new phrases and new personal data by educators.

Below those enhancements are explained in more detail:

1. Individual Memory - IPM
   When talking to people with dementia or to children, it is essential to "know" details about the life and the environment of the person to whom one is talking. We added such a basic memory, but still have to refine it (the data structure and the bot program processing it). AIML 2.0 allows defining new tags. To use IPM, we added 4 new tags to write new

individual information into IPM, or to read from IPM. We introduced one tag to write information, and two tags, which read information from the IPM: one tag leads to reading randomly, the other selectively.
Another tag checks, whether a information selected is available.

2. Multi-client properties
We have decided to implement in a first step a central architecture based on a server. So we are able to learn about the system's deficits, and the needs to improve the system. To gain flexibility, we will implement multi-client properties, to serve more than one client simultaneously.

3. Timeout
Bot and database are modified, to be able to start a dialog, or to continue talking, even if the human does not answer.

4. Editor tools
There are tools available to edit AIML, e.g. GaitoBot AIML Editor. (GaitoBot, 2015). We have to investigate those tools and select one, which is suitable to be used by educators. In addition, we need editor functions to edit the individual personal memory.
If none of the available editors is appropriate, we will write an Habláme specific editor.

6 RESULTS

We have implemented a prototype which is able to talk within a limited domain to an adult person. So the text to speech component, speech to text, the server, enhanced AIML, the timeout function, and the proper function of the database can be demonstrated.

Three major tasks have yet to be accomplished:

1. Complete the system and add
   - multi-client properties, and an
   - editor for educators.
2. Next step is to complete the AIML patterns so that children are motivated to talk to the system (current status: project already started).
3. Test the system together with educators with healthy children first, and then with children with atypical development with respect to its educational benefits (current status: project will start in summer)

ACKNOWLEDGEMENTS

This work had been partially funded by the EU Project GAVIOTA (DCI-ALA/19.09.01/10/21526/ 245-654/ALFA 111(2010)149).

REFERENCES

Balandin, R., Hegwein, M., Barthel, A., 2014. Erweiterung und Analyse eines Chatbot. Project Report, HAW Würzburg-Schweinfurt (FHWS).

Bossavit, B., Ostiz, M., Lizaso, L., Pina, A., Grötsch, E., 2014. Software with Augmented Reality, Virtual Reality, and Natural Interaction for Atypical Development Individuals / Software con realidad avanzada, realidad virtual e interacción natural para personas de desarrollo no típico. In: Fernández, M.I. (Ed.) Augmented Virtual Realities for Social Development – Experiences between Europe and Latin America / Realidades Virtuales aumentadas para el desarrollo social – Experiencias entre Europa y Latinoamérica, pages 103 – 129, Universidad de Belgrano, Buenos Aires, 2014, ISBN 978-950-757-047-6.

ELIZA, 2013.
www.med-ai.com/models/eliza.html (Jan 7, 2015).

GaitoBot, 2015.
www.gaitobot.de/gaitobot/ (March 25, 2015).

Grötsch E., Pina A., Schneider M., Willoweit B., 2013. “Artificial Communication“ - Can Computer Generated Speech Improve Communication of Autistic Children?. In Proceedings of the 5th International Conference on Computer Supported Education, Aachen, pages 517-521
DOI: 10.5220/0004412805170521.

JEOPARDY, 2011.
http://www.nytimes.com/2011/02/17/science/17jeopardy-watson.html?_r=0, (Jan 6, 2015).

Schmitt, S., 2014. Ein plattformunabhängiges Dialogsystem für gesprochene Sprache, Bachelor thesis, HAW Würzburg-Schweinfurt (FHWS).

UIMA, 2013.
http://uima.apache.org/ (Jan 6, 15).

Wallace, R., 2014. The Anatomy of A.L.I.C.E. http://www.alicebot.org/documents/Anatomy.doc. (Jan. 6, 15).

Weizenbaum, J., 1966. ELIZA - A Computer Program for the Study of Natural Language, Communication between Man and Machine. Communications of the ACM. New York 9.1966,1. ISSN 0001-0782.