

Extension and Prerequisite: An Algorithm to Enable Relations Between Responses in Chatbot Technology

Abbas Saliimi Lokman and Jasni Mohamad Zain
Faculty of Computer Systems and Software Engineering,
University Malaysia Pahang, Lebuhraya Tun Razak, 26300 Kuantan, Pahang, Malaysia

Abstract: Problem statement: Artificial intelligence chatbot is a technology that makes interactions between man and machines using natural language possible. From literature, we found out that in general, chatbot are functions like a typical search engine. Although chatbot just produced only one output instead of multiple outputs/results, the basic process flow is the same where each time an input is entered, the new search will be done. Nothing related to previous output. This research is focused on enabling chatbot to become a search engine that can process the next search with the relation to the previous search output. In chatbot context, this functionality will enhance the capability of chatbot's input processing. **Approach:** In attempt to augment the traditional mechanism of chatbot processes, we used the relational database model approach to redesign the architecture of chatbot in a whole as well as incorporated the algorithm of Extension and Prerequisite (our proposed algorithm). By using this design, we had developed and tested Virtual Diabetes physician (ViDi), a web-based chatbot that function in specific domain of Diabetes education. **Results:** Extension and prerequisite enabled relations between responses that significantly make it easier for user to chat with chatbot using the same approach as chatting with an actual human. Chatbot can give different responses from the same input given by user according to current conversation issue. **Conclusion:** Extension and prerequisite makes chatting with chatbot becomes more likely as chatting with an actual human prior to the relations between responses that produce a response related to the current conversation issue.

Key words: Chatbot, keywords matching, conversation topic, response relation, response selection

INTRODUCTION

In 1950, mathematician Turing (1950) proposed the question "Can machines think?". Since then, a number of technologies have been emerged in computer science field by the attempt to encounter that particular question that generally founded the field of Artificial Intelligence. One of technology that attempts to visualize an intelligence machine is chatbot or chatter robot that makes interaction between man and machine using natural language possible. First introduced by Weizenbaum (1966) (an MIT professor), chatbot ELIZA then famously became an inspiration for computer science and linguistic researchers in creating a computer application that can understand and response to human language. The huge breakthrough in chatbot technology came in 1995 where Dr. Richard Wallace, an ex-Professor of Carnegie Mellon University combine his background in computer science with his interest in the internet and natural language processing to produce Artificial Linguistic

Internet Computer Entity (ALICE). ALICE that later being described as a modern ELIZA is a three times winner of Loebner's annual instantiation of Turing's Test for machine intelligence (Shah, 2006). When computer science evolves, so does the chatbot technology. As for a chatbot that need to have a huge knowledge-based which some call it as a "chatbot's brain", managing data is really critical. Reviewing the evolving of chatbot technology that paralleling with the evolving of computer science technology, ELIZA stored its knowledge-based data by embedding it right into the code and later came ALICE that uses Artificial Intelligence Markup Language (AIML) which is a derivative of Extensible Markup Language or XML (Shawar and Atwell, 2007; Wallace, 2009) to stored the knowledge-based data. Then with Relational Database Model together with Database Management System (DBMS) mechanism, came chatbots that taking an advantage of it. One of an example is VPbot, an SQL-Based chatbot for medical application (Ohno-Machado and Weber, 2005). Developed by Ohno-Machado and

Corresponding Author: Abbas Saliimi Lokman, Faculty of Computer Systems and Software Engineering,
University Malaysia Pahang, Lebuhraya Tun Razak, 26300 Kuantan, Pahang, Malaysia

Weber (2005) from Harvard University, VPbot is a chatbot that takes advantage of a Relational Database Model to stored, manage and even used SQL language (database scripting language) to perform the chatbot main process that is keywords/pattern matching.

Extension and Prerequisite are proposed to enable relations between responses in chatbot technology. Designed by using an approach of Relational Database Model, Extension and Prerequisite is implemented both in keywords matching process and knowledge-based authoring process. Currently chatbots are designed to response for user's input in a one-way input-response mechanism without any parameter that holds the conversation issue. It was like a search engine paradigm where user typed an input and engine will produce an output based on that input alone. Then if a new search parameter is being entered, the search process will start all over again without any relation to the previous search. Therefore, in general chatbot process model (Fig. 1), input 1 will return a response 1 and so on until input n will return a response n (last input from user).

Although there is a used of some technique that will hold the "topic" of the conversation, AIML <that> or <topic> tag and VPbot topic parameter (Ohno-Machado and Weber, 2005), those technique does not exactly hold the conversation issue because the "topic" mechanism simply is a technique that replaces word/phrase with another word/phrase that had been stored as a constant variable at that particular conversation. There also a statement that suggested the irrelevantly of the response given by chatbot (AIML chatbot) prior to the conversation issue. Jia (2004) stated that within two or more rounds, most users could find that the responses from the chatbot are stupid and irrelevant with the topic and the context. Shawar and Atwell (2005) stated that there is a logical inconsistencies in chatbot replies given by example that previous chatbot sentence suggested a non-enthusiasm about sport but later become enthuses about football. In other words, "topic" is basically used as a replacement over pronoun with constant noun. Example conversation implementing "topic" mechanism is; "I broke my hand", "Did it hurt?", replacement of pronoun "it" to the constant noun "hand".

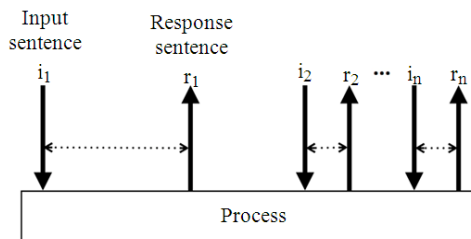


Fig. 1: General chatbot process model

Whereby, the real matter in holding conversation issue is a hypothetical conversation lines that human draw when they had a conversation with one another. For example, as human talk about "car", "What brand is your car?" "Toyota", "How much is it?" "A thousand Dollar" and later changed to the issue of "house" within the same conversation, "Do you live in a bungalow?" "Yes", "Where is it located?" "Near the beach". From this example, there is hypothetical line regarding relations between responses in the issue of "car" and another line in the issue of "house". This line is basically a connection that being created in human conversation from a responses that relate to each other. Extension and Prerequisite is focus on creating this line in a human conversation with chatbot.

MATERIALS AND METHODS

To test an algorithm of Extension and Prerequisite, we designed and developed a chatbot named Virtual Diabetes physician (ViDi), a web-based chatbot that being developed to responsible for specific domain of Diabetes education. Taking advantage of Relational Database Model approach, we redesign the whole architecture of chatbot with Extension and Prerequisite algorithm being incorporated into it. In technical details, ViDi is being coded using Hypertext Preprocessor (PHP) programming language together with Asynchronous Javascript + XML (AJAX) technology which contains a technology of Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), XMLHttpRequest (XHR) and Document Object Model (DOM) that being accessed via JavaScript. As for database, ViDi uses MySQL database with phpMyAdmin as a Database Management System (DBMS). Figure 2-4 displays ViDi's UI (User Interface) design. Figure 2 is a chatting UI for users while Fig. 3 and 4 are knowledge-based (responses and keywords) management UI known as vBrain for authors. Note that ViDi is a Bahasa Malaysia human language chatbot (Lokman and Zain, 2010) and that being the case, the contents presented in each UI are mostly originated from this language. Nevertheless, English language is being used for interactions between authors and ViDi regarding vBrain functionality. On that account, authors with the knowledge of English language can just changed ViDi's knowledge-based data to any language within the used of Latin/Roman alphabet to make it functions according to their desired language.

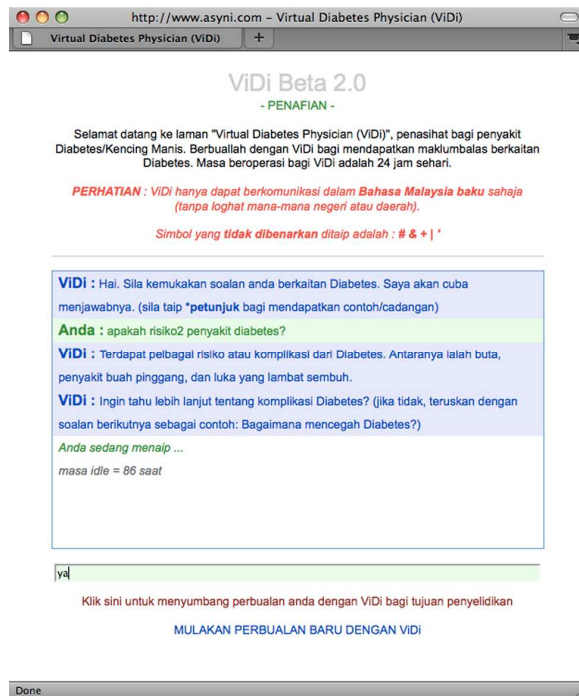


Fig. 2: ViDi chatting interface

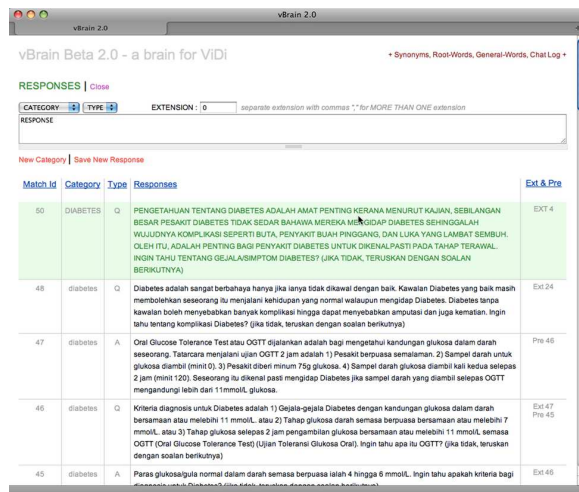


Fig. 3: vBrain-managing ViDi's response

Extension and Prerequisite are designed to study with each other. In all responses data, we put another two variables named Extension and Prerequisite to store the unique Response/Match ID (note that every response have one). The default value for each variable are "0" and will change if author of ViDi's knowledge-based linked that particular response with other response/s (can be linked to more than one response). The process of linking responses is done at vBrain UI. In vBrain UI of adding new response (Fig. 5) or edit

current response, there is an input box for Extension data. In that box, author can add as many Response ID as they want (must be separate by commas ",") to be the Extension data for that particular response. After saving the new/edited response data (let say Response 1), vBrain will automatically define the Prerequisite data for each Response ID in Response 1 Extension data to be linked back to Response 1. In short, after a new response being add or edit, the changes value in Extension input box will define the relation between that particular response to other response/s by creating a forward (for Extension) link to every Response ID in Extension data and a back (for Prerequisite) link from every Response ID in Extension data back to the original response. As Extension data can hold as many Response ID as the author want, the relations between responses can be one way, recursive or pointing back to previous response. Figure 6 show a visual explanation regarding relations' possibilities in responses database. Abbreviations note:

rid = Response ID; ext = Extension; pre = Prerequisite. Symbols note: solid line = Extension link; dotted line = Prerequisite link

Referring back to Fig. 4, each selected response will display their Extension and Prerequisite value and author can simply click on them to jump into that particular response. Figure 4 also shown a list of keywords for the selected response and the last set of keywords is named "Prerequisite keywords" that is a keywords that ViDi will find if the selected Response ID is in the Extension data of current ViDi's response. Implementing Extension and Prerequisite, the algorithm steps regarding chatbot processes (receive input from user, performing keywords matching process and generate response back to user) became as follows:

1. From previous response-chatbot will hold previous response's Extension data (if any)
2. Receive another input from user
3. Processing input (normalization, synonyms replacement and so on)
4. Analyzing Extension data (one or more next response's ID)
5. Keywords matching regarding Extension ID/s (one by one if many IDs)
6. If match, generate response and hold new response's Extension data (if any)
7. If no match, run keywords matching process for the entire keywords database
8. If match, generate response and hold new response's Extension data (if any)
9. If no match, generate response for user to enter another input. Hold same Extension data as Step 1

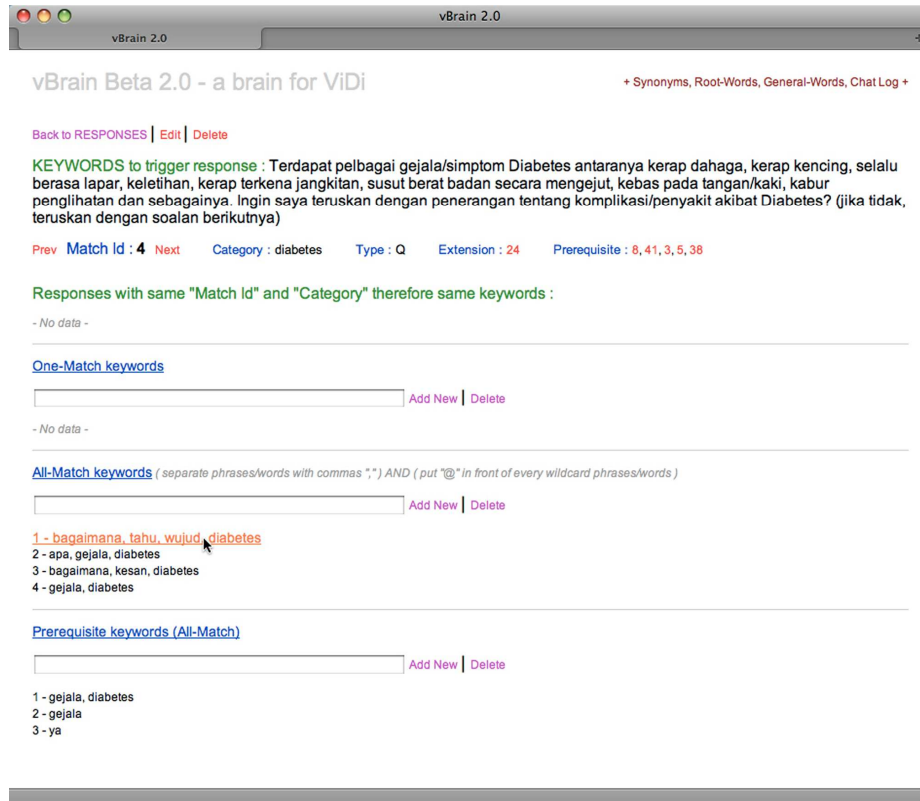


Fig. 4: vBrain-managing ViDi's keywords, extension and prerequisite data for each response



Fig. 5: vBrain-add new response UI

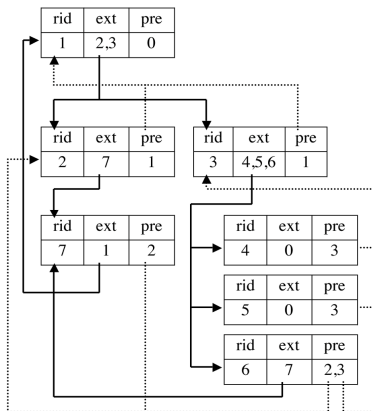


Fig. 6: Sample relations between responses

RESULTS

For testing parameters, we used ViDi's knowledge of Diabetes disease as a domain case study. Although the test is in specific domain of knowledge, the relation's architecture was rather similar, as it is not based on any specific knowledge, but on relations between responses in any chatting conversation. Figure 7 and 8 shows sample conversations between user and ViDi regarding the implementation of Extension and Prerequisite algorithm. As mention earlier, ViDi is a Bahasa Malaysia human language chatbot and therefore, this language represents all conversation utterances in Fig. 7 and 8. Hence, all conversation utterances will be translated into English human language as this research's representative language.



Fig. 7: Sample conversation with ViDi (Result 1)



Fig. 8: Sample conversation with ViDi (Result 2)

The translation for Fig. 7 conversation is as follows:

User (green field): “what types of diabetes?”
ViDi (blue field): “Diabetes is divided into 3 types that are Type 1 (insulin dependent Diabetes), Type 2 (non insulin dependent Diabetes) and Gestational Diabetes (Diabetes during pregnancy).”
ViDi: “Want to know what is insulin? (if not, proceed with next question)”
User: “yes”
ViDi: “Insulin is a hormone...”-A description about insulin.
ViDi: “Want to know more about insulin? (if not, proceed with next question)”
User: “yes”
ViDi: “When we eat,...”-A description about how insulin study.

The translation for Fig. 8 conversation is as follows:

User (green field): “Tell me about type of diabetes”
ViDi (blue field): “Diabetes is divided into 3 types that are Type 1 (insulin dependent Diabetes), Type 2 (non

insulin dependent Diabetes) and Gestational Diabetes (Diabetes during pregnancy).”

ViDi: “Want to know what is insulin? (if not, proceed with next question)”

User: “also can”

ViDi: “I did not understand the meaning of (also can). Please give a different response.”

User: “Oh ok..”

User: “Yes... tell me about insulin”

ViDi: “Insulin is a hormone ...”-A description about insulin.

ViDi: “Want to know more about insulin? (if not, proceed with next question)”

Figure 7 (Result 1) and Fig. 8 (Result 2), both sample conversations demonstrate users chatting experience with chatbot that become more likely as chatting with an actual human than chatbots without the implementation of Extension and Prerequisite algorithm. The justification for that statement is by looking back to current chatbot’s process mechanism, an input is being process by the entire keywords database and therefore, a same input may result in the same output (maybe different if the algorithm makes a random selection from several outputs but it was all in the same semantic category). In implementing Extension and Prerequisite, same input can produce a different output relating to previous output. Result 1 shows that input “yes” could produce a different output relating to previous output that asking about if you want to know about insulin (question 1) and if you want to know more about insulin as addition to the general description answered for question 1 (question 2). Both questions 1 and 2 provide semantically different answer but require the same keyword. Referring to casual human chatting conversation, human was tending to give a short input (elliptic input) rather than a long input just to makes a good or grammatically correct sentence. For example related to Result 1, if the question is “Want to know what is insulin?”, human did not usually answer it as “Yes, i want to know about insulin”. Same as the second question “Want to know more about insulin?” did not answered as “Yes, tell me more about insulin”. Unfortunately, this kind of answer are what current chatbot technology need for user to provide in order to be understandable because chatbot needs a sufficient keywords to differentiate on which questions did the user actually answered. Back to casual human chatting conversation, for both questions, answering “yes” is just sufficient enough to make it an understandable answers and that it what Extension and Prerequisite successfully enabled in chatbot technology. Result 2 demonstrated an additional capability of

Extension and Prerequisite. Referring back to chatbot with Extension and Prerequisite implementation algorithm steps in previous section, Step 9 clarify that “If no match, generate response for user to enter another input. Hold same Extension data as Step 1”. As such, Result 2 demonstrated that chatbot did not understand user’s first response, therefore asking the user to enter another input while still holding the same previous Extension data. Then, the next input from user will again being processed in the precedence of Extension data first before looking out in the entire keywords database for a match.

DISCUSSION

From the presented results, Extension and Prerequisite had successfully enabled relations between responses in chatting conversation between human and chatbot. Figure 9 shows an extended version of general chatbot process model from Fig. 1 with the relation between responses created by the implementation of Extension and Prerequisite algorithm. The relations created by this algorithm is a specific interaction between responses that relate to each other in the context of a whole sentence, not as the “topic” mechanism that basically used as a replacement over pronoun with constant noun. This relation opens the possibly for chatbot to have a conversation that can hold the specific issue per interest of the conversation (one conversation can have multiple and different issues). Concerning to the specific issue being held in a conversation, keywords matching (for chatbot) and input entering (for human) become more productive.

For chatbot, implementing Extension and Prerequisite will reduce the processing time for keyword matching process because the algorithm will narrow down the size of keywords to be match which is from an entire keywords database, to just a keywords database concerning the Extension data. As for human, input entering become much simpler when they can just entered elliptic input for chatbot to understand.

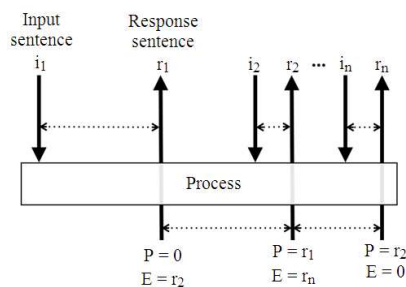


Fig. 9: Chatbot process model with response’s relation

Elliptic input is an essential concern in chatbot processes because it was a general habit in human chatting approach to response in a manner of remembering the current conversation issue and give response in a same manner. This issue is impartially related to a study in human cognitive function that suggested a “Long-Term Working Memory” in human performing cognitive task given by example when human reading a sentence in a text, they must have access to previously mentioned actors and objects and also a contextual information (in that particular subject) in order to integrate coherently with the information presented in the current sentence (Ericsson and Kintsch, 1995). This relation also supported the issue of the needs for instructional approaches rather than minimally-guided instruction that less effective and less efficient on a guidance for the human learning process (Kirschner *et al.*, 2006). In the context of chatbot that functions as knowledge representation system, relations between responses can be greatly used for the guided-chatting activity as presented in Results in which chatbot keep giving guidance on how to proceed. This chatting scenario will principally eliminate the idle time when users did not know what to response and later leaving the chatbot as they become bored.

The implementation of guided-chatting activity was first came to our concern when we run the public beta trial for ViDi. During this preliminary trial, we got feedbacks from users saying that they generally did not know what to say to ViDi even though they want to know more about Diabetes. This mostly because they did not perceived on what they should know about Diabetes disease. For that being the case, we incorporate the instruction utterance at the end of chatbot’s responses to be the guidance for users on how to proceed (users can choose to ignore them). As the implementation and testing stage are recursive, we got positive feedbacks regarding this particular implementation.

CONCLUSION

In this research, an additional algorithm for chatbot technology is proposed. The algorithm named Extension and Prerequisite had enabled chatbot to have relations between responses that open up a possibility for chatbot to have a specific issue conversation in a more controlled approach. This functionality makes a chatting activity with chatbot become more productive for human and chatbot itself prior to the focus issue being the main concern. As a result, elliptic inputs by users become understandable by chatbot and processing time regarding finding a keywords match in chatbot

process become faster. The relations between responses also open the possibilities for chatbot to function as a diagnostic systems in which several questions relate to each other that finally produced a final output regarding the diagnosis result. Starting with the first main question, next question will be determined depending on user's answer and goes on until the end of diagnosis, forming a pyramid of questions that is a final diagnosis results at the bottom of the pyramid describing by different path taken will produce different end result. For this research, ViDi version 2 is being used as testing material. The earlier version of ViDi (ViDi version 1) although being slightly different from ViDi version 2 as Diabetes Physician (ViDi version 1 functions as Diabetes Dietitian), it also implementing algorithm regarding enabling relations between responses (Lokman and Zain, 2009). Named "Vpath", this parameter will be used to determine conversation path that can lead to final advice on user's diet. Vpath however did not directly create relations between responses but rather being used to determine which path did the conversation take during question answering session. For that function, Vpath become limited to holding only one path per conversation session. On that account, Extension and Prerequisite algorithm is being proposed to makes relations between responses became more flexible.

REFERENCES

- Ericsson, K.A. and W. Kintsch, 1995. Long-term working memory. *Psychol. Rev.*, 102: 211-245. PMID: 7740089
- Jia, J., 2004. The study of the application of a web-based chatbot system on the teaching of foreign languages. *Proceeding of the Society for Information Technology and Teacher Education International Conference, (SITE'04)*, AACE Press, USA., pp: 1208-1211. <http://www.editlib.org/p/13634>
- Kirschner, P.A., J. Sweller and R.E. Clark, 2006. Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential and inquiry-based teaching. *Educ. Psychol.*, 41: 75-86. DOI: 10.1207/s15326985ep4102_1
- Lokman, A.S. and J.M. Zain, 2009. An architectural design of Virtual Dietitian (ViDi) for diabetic patients. *Proceeding of the 2nd IEEE International Conference on Computer Science and Information Technology*, Aug. 8-11, IEEE Xplore Press, Beijing, pp: 408-411. DOI: 10.1109/ICCSIT.2009.5234671
- Lokman, A.S. and J.M. Zain, 2010. Chatbot enhanced algorithms: A Case study on implementation in Bahasa Malaysia human language. *Networked Digit. Technol.*, 87: 31-44. DOI: 10.1007/978-3-642-14292-5_5
- Ohno-Machado, L. and G.M. Weber, 2005. *Data Representation and Algorithms for Biomedical Informatics Applications*. 1st Edn., Harvard University, Cambridge, MA., USA., ISBN: 0-542-11988-9, pp: 184.
- Shah, H., 2006. ALICE: An ACE in digitaland. *tripleC*, 4: 284-292. <http://triple-c.at/index.php/tripleC/article/view/46/44>
- Shawar, B.A. and E. Atwell, 2005. A chatbot system as a tool to animate a corpus. *ICAME J.*, 29: 5-24. <http://icame.uib.no/ij29/ij29-page5-24.pdf>
- Shawar, A.B. and E. Atwell, 2007. Chatbots: Are they really useful. *LDV-Forum Band*, 22: 31-50.
- Turing, A.M., 1950. Computing machinery and intelligence. *Pars. Tur. Test*, 49: 433-460. <http://www.cs.umbc.edu/courses/471/papers/turing.pdf>
- Wallace, R.S., 2009. The Anatomy of A.L.I.C.E. *Pars. Tur. Test*, 3: 181-210. DOI: 10.1007/978-1-4020-6710-5_13
- Weizenbaum, J., 1966. ELIZA-a computer program for the study of natural language communication between man and machine. *Commun. ACM*, 9: 36-45. DOI: 10.1145/365153.365168