

Chatbots are natural web interface to information portals

Bayan Abu Shawar
Information Technology and Computing Department
Arab Open University, Jordan
b_shawar@aou.edu.jo

Abstract

In this paper we describe a way to access information using chatbot, without the need for sophisticated natural language processing or logical inference. FAQs are Frequently-Asked Questions documents, designed to capture the logical ontology of a given domain. Any Natural Language interface to an FAQ is constrained to reply with the given Answers, so there is no need for NL generation to recreate well-formed answers, or for deep analysis or logical inference to map user input questions onto this logical ontology; simple (but large) set of pattern-template matching rules will suffice. In this paper and as an evidence for this argument, the FAQ in the School of Computing (SoC) at the University of Leeds as well as Perl, Linux, and Python were used to retrain the ALICE chat-bot system, producing FAQchat. The replies from FAQchat looks like results generated by WWW search engines such as AskJeeves or Google. User trials with AskJeeves, Google and FAQchat demonstrate that FAQchat is a viable alternative, and in fact many users prefer it to Google as tool to access FAQ databases. The restricted domain of an FAQ is special case of Question-Answering which does not require the sophisticated analysis techniques.

1. Introduction

An Information Portal is a website which provides information in the form of semi-structured text; a widely-used format is the FAQ website, where information is presented as a set of Frequently-Asked Questions in plain English (or other natural language), and the answers to these questions. An FAQ website provides information on a specific topic, and the authors are typically experts who have had to answer questions on the topic (e.g. helpdesk managers, instructors) and want to comprehensively cover all likely questions so as not to have to answer these interactively (face-to-face) in future. The standard “interface” to an FAQ is not a special-purpose browser or

a natural-language front end, but just a Table of Contents and/or Index. Users are typically invited to wander through the FAQ document (using their normal web-browser) till they find the answer to their question; arguably FAQs are really Frequently sought Answers each annotated with a typical Question. Browsing the entire web-page or web-site is fine for limited FAQs, but gets less manageable for larger domains, which may be hierarchically organised. For example, the online FAQ for the Python programming language has sub-sites several sub-pages for Python subtopics, and also sub-sites for English, Spanish, and other translations, so users have to navigate a hierarchical ontology.

A chatbot is a conversational software agent, which interacts with users using natural language. The idea of chatbot systems originated in the Massachusetts Institute of Technology [13], where Weizenbaum implemented the ELIZA chatbot to emulate a psychotherapist. Then PARRY [10] was implemented to simulate a paranoid patient. “Colby regarded PARRY as a tool to study the nature of paranoia, and considered ELIZA as a potential clinical agent who could, within a time-sharing framework, autonomously handle several hundred patients an hour.” [14]. Nowadays several chatbots are available online, and are used for different purposes [19] such as: MIA [15] which is a German advisor on opening a bank account; Sanelma [61] a fictional female to talk with in a museum that provides information related to specific piece of art; Cybelle [17], and AskJeeves [8], a web-based search engine. In this paper the chatbot is used as a natural language interface to search WWWFAQ pages.

ALICE/AIML chatbot architecture and how the chatbot was trained by dialogue corpora then turned to be trained with FAQ domain are presented in section 2. Section 3 describes how the program was adapted to deal with WWWFAQs and the problems we faced. Section 4 shows the results and samples of the chatting with the FAQchat. Section 5 displays a comparison between AskJeeves, Google and FAQchat. The conclusion that shows that FAQchat could be used as an open-ended question answering systems using WWWFAQ pages is presented in section 6.

2. ALICE/AIML chatbot architecture

We have worked with the ALICE open-source chatbot initiative. ALICE [1], [12] is the Artificial Linguistic Internet Computer Entity, originated by Wallace in 1995. A sample of chatting with ALICE is shown in figure 1. ALICE stores knowledge about English conversation patterns in AIML files. AIML, or Artificial Intelligence Mark-up Language, is a derivative of Extensible Mark-up Language (XML). It was developed by the Alicebot free software community during 1995-2000 to enable people to input dialogue pattern knowledge into chatbots based on the ALICE free software technology.

User: Hello, my name is George and I am the human.

ALICE: Hi George.

User: Would you like to play a game?

ALICE: Sure.

User: Ok then. I will email it to you by use of the CD drive.

ALICE: What hobbies do you enjoy?

User: I like cinema.

ALICE: What types of films?

User: All.

ALICE: What movie did you see this year that you liked?

User: I don't have time for movies.

ALICE: I thought you liked cinema.

User: I did, but then you started to say stupid things.

Figure 1. A sample of conversation with ALICE (from Loebner prize contest1 2004)

AIML consists of data objects called AIML objects, which are made up of units called topics and categories. The topic is an optional top-level element; it has a name attribute and a set of categories related to that topic. Categories are the basic units of knowledge in AIML. Each category is a rule for matching an input and converting to an output, and consists of a pattern, which matches against the user input, and a template, which is used in generating the Alice chatbot answer.

The AIML pattern is simple, consisting only of words, spaces, and the wildcard symbols `_` and `*`. The words may consist of letters and numerals, but no other characters. Words are separated by a single space, and the wildcard characters function like words. The pattern language is case invariant. The idea of the pattern matching technique is based on finding the best, longest, pattern match.

3. Retraining ALICE with different corpora

We have techniques for developing new ALICE language models, to chat around a specific topic: the techniques involve machine learning from a training corpus of dialogue transcripts, so the resulting chatbot chats in the style of the training corpus [2], [3], [4], [5], [7]. For example, we have a range of different chatbots trained to chat like London teenagers, Afrikaans-speaking South Africans, loudmouth Irishmen, etc by using text transcriptions of conversations by members of these groups. The training corpus is in effect transformed into a large number of categories or pattern-template pairs. User input is used to search the categories extracted from the training corpus for a nearest match, and the corresponding reply is output. This simplistic approach works best when the user's conversation with the chatbot is likely to be constrained to a specific topic, and this topic is comprehensively covered in the training corpus. This should definitely be the case for a chatbot interface to an FAQ, a Frequently-Asked Questions document.

The logic of chatbot question-answering is built into an FAQ document by the designer. The designer specifies the taxonomy of possible Answers; whatever Question a user may pose, the chatbot can only reply with one or more Answers from this taxonomy, as the topic is comprehensively defined by this ontology. This suggests that sophisticated Natural Language Processing analysis used in systems like AskJeeves is redundant and pointless in an FAQ-query chatbot. Querying an FAQ is more like traditional Information Retrieval: a user query has only to match one or more documents (Answers) in the document set. However, users may prefer to pose a query as a Natural Language question rather than a Google-style list of keywords; so they may yet prefer a chatbot interface to an FAQ over Google-style traditional information retrieval. AS an evidence for this case, the FAQ in the School of Computing (SoC) at University of Leeds was used to retrain the ALICE chatbot system, producing FAQchat. Users trials with AskJeeves, Google and FAQchat demonstrate that FAQchat is a viable alternative, and in fact many users prefer it to Google as a tool to access FAQ databases.

4. Creating an open-ended FAQchat

The WWW FAQs are structured databases; most of the "data-cleaning" problems found with analysis of spoken dialogue corpora such as overlapping, and more than two speakers are not found in the FAQs. Moreover almost all HTML tags are recognized by the ALICE interpreter because the AIML definition allows HTML tags to be embedded within templates. The questions and answers were extracted from the HTML files of the FAQ. However, each FAQ website has different annotations for questions/answers, and links as illustrated in the next sub section. For this reason, different versions of the program were developed to deal with the different annotations.

¹ <http://www.loebner.net/Prizef/loebner-prize.html>

4.1 WWW FAQs format and problems

In addition to the previous work generated on the FAQ of the School of Computing (SoC²) at University of Leeds, we use other FAQs: Perl³, Linux⁴ and Python⁵. In general each web site uses different tags to denote questions and answers. The list of these links differs from one site to another. Figure 3 shows sample of Perl FAQs, how these tags are used to denote questions and answers in selected websites are presented in table 1.

Within the SoC⁶ FAQs the following annotations are used:

- Some questions are marked by: `<DIV CLASS="sect1">` tag and the answers by: `<p>`. The problem is when to consider the `<p>` tag as a part of the question, or when it denotes the beginning of the answer.
- Some questions are marked by: `<DIV CLASS="question">` tag and the answer by: `<DIV CLASS="answer">`.

In general, the following problems arise from the different WWW FAQs:

- No standard annotation for references or questions/answers.
- Extra annotations represented by the beginning of each file and the footer.
- Using special character entities denoted by the “&” sign; these are not allowed in AIML.

Because of these problems different versions of the program were developed to generate the AIML files.

4.2 Processing the FAQ websites

FAQchat does not need a linguistic knowledge module, and also in principle is language independent: it can be trained with FAQs in any natural language. The Java program we developed and used before was adapted to deal with the WWWFAQ. The way FAQchat works is described below:

1. All questions and answers are extracted from the whole database after applying a filtering process to remove unnecessary tags.

² <http://www.comp.leeds.ac.uk/faq/>

³ <http://perldoc.perl.org/index-faq.html>

⁴ <http://www.faqs.org/contrib/linux/Linux-FAQ/>

⁵ <http://www.python.org/doc/faq/>

⁶ <http://www.comp.leeds.ac.uk/faq/>

```
<p><a name="__index__"></a></p>
<!-- INDEX BEGIN -->
.....
<ul><li><a href="#what_is_perl">What is Perl?</a></li>
<li><a href="#what_are_perl4_and_perl5">What are
perl4 and perl5?</a></li>
<li><a href="#what_is_ponie">What is Ponie?</a></li>
....</ul><!-- INDEX END -->
.....
<h2><a name="what_is_perl">What is
Perl?</a></h2>

<p>Perl is a high-level programming language with an
eclectic heritage written by Larry Wall and a cast of
thousands. It derives from the ubiquitous C program
.....</p>
```

Figure 3. A sample of Perl FAQs

2. The FAQ database yielded questions and (Answers). A list of links is constructed, containing the links from FAQ to web pages containing answers.
3. A dictionary is created, containing all words in the questions with frequencies of occurrence. Then the first and second most significant words (least frequent words) are extracted from each question.
4. AIML pattern-matching rules, known as “categories”, are created. There are two possible types of match: input matches a complete FAQ questions; or input matches 1st or 2nd most significant word in an FAQ question (least frequent words). There are two types of responses generated, which either has the direct answer (in the instance where only one match was found) or if the most significant words are found in more than one question, multiple links are returned as a reply.

For example: assume FAQ database contains the followings:

```
<h2><a name="what_is_perl">What is Perl?</a></h2>

<p>Perl is a high-level programming language with an
eclectic heritage written by Larry Wall and a cast of
thousands. It derives from the ubiquitous C program
.....</p>
```

Then at least two types of categories will be created:

Atomic category:

```
<category>
```

```
<patter> WHAT IS PERL</pattern>
```

```
<template> Perl is a high-level programming language
with an eclectic heritage written by Larry Wall and a cast
of thousands. It derives from the ubiquitous C program
</template></category>
```

FAQ site	Tag used to denote link references	Tags used to denote question	Tags used to denote answers
Perl	<!-- INDEX BEGIN--> <!-- INDEX EN--.>	<h2> and </h2> denotes a question begin and end	<p> and </p> denotes an answer begin and end
Linux	<DIV CLASS "QADASET">	<DIV CLASS "QUESTION"> Q:	<DIV CLASS "ANSWER"> A CLASS "QANAENTRY" denotes end
Python	<div class="contents topic" id="contents">	<div class="section" id="what-is-python"> <h5>..</h5>	<p> denotes answer. "<div class ="section" ..> denotes the end

Table 1. WWW FAQs tags

One of the default categories is:

<category>

<patter> WHAT*PERL</pattern>

<template> Perl is a high-level programming language with an eclectic heritage written by Larry Wall and a cast of thousands. It derives from the ubiquitous C program you will find more in the link:.....</template>
</category>

5. Comparing Google, AskJeeves and FAQchat

Google is “a search engine which is very easy to use. It returns pages based on the number of sites linking to them and how often they are visited, indicating their popularity.” [9] Search engines like Google retrieve information in four phases [8]:

1. Obtaining documents to be searched. The method used gives a classification of search engine types:
 - Search engines which use crawlers, or spiders to get URLs such as Google;
 - Search engines based on human submission;
 - Others that are a combination of the two.
2. Preparing the documents to be searched, which involve operations such as: filtering the text, and extracting the meaningful items.

3. Indexing the items. One of the mechanisms used by Google is the inverted file structure. Three stages are applied here:
 - Each document has a unique ID;
 - A dictionary of all stemmed words from all documents is created.
 - Each item in the dictionary is associated with a pointer to the inversion list. The inversion list associates each item to all documents containing it.
4. The matching process to give the best answer to a specific user query. One of the most widely used methods is the vector space model, where a two-dimensional array (term by document) is constructed with size $M \times N$; M represents the items in dictionary, and N represents the documents. A weighting scheme may be applied such as column normalisation or tf-idf. The users query is represented as a vector of size M , and it is normalised, stemmed, and weighted in the same manner as the document’s items. At the end the best hit will be selected using different methods of ranking. This ranking algorithm is the “hidden ingredient” differentiating rival search engines.

Most search engines break up the user query into keywords, and return results according to keyword matches like Google. AskJeeves [6] is a search engine that returns a result after understanding the query, using a question-processing engine to understand the meaning of the words and grammar of the question. When a user inputs a question, AskJeeves returns a list of possible questions it thinks the user could be asking; the user then

selects a target question, and AskJeeves returns web-sites that answer the target question.

In the ALICE architecture, the “chatbot engine” and the “language knowledge model” are clearly separated, so that alternative language knowledge models can be plugged and played. Another major difference between the ALICE approach and other chatbot-agents such as AskJeeves is in the deliberate simplicity of the pattern-matching algorithms: whereas AskJeeves uses sophisticated natural language processing techniques including morphosyntactic analysis, parsing, and semantic structural analysis, ALICE relies on a very large number of basic “categories” or rules matching input patterns to output templates. ALICE goes for size over sophistication: it makes up for lack of morphological, syntactic and semantic NLP modules by having a very large number of simple rules. The default ALICE system comes with about fifty thousand categories, and we have developed larger versions, up to over a million categories or rules.

We tried to see if sophisticated NLP tool, exemplified by AskJeeves, is better than keyword-based IR, exemplified by Google, for accessing FAQ. We designed set of typical English-language questions for the School of Computing FAQ, and posed these to both search-engines, constrained to search only the SoC FAQ website. The correct answer was included in 53 percent of AskJeeves answers, and 46 percent of Google answers, which indicated no significant difference in performance: sophisticated NLP is not better than word-based pattern matching when answering questions from restricted-domain FAQ.

However, users may prefer to input natural language questions rather than keywords, so we also asked set of users to compare Google and FAQchat. We set series of information-gathering tasks, and set up an interface which allowed users to type in question; this was sent to both FAQchat and Google, with responses displayed side-by-side. We found that 68 percent of FAQchat answers were considered correct by users, but only 46 percent of Google answers were correct. Users were also asked which system they preferred overall: 47 percent preferred FAQchat, while only 11 percent preferred Google [6].

The aim of this evaluation is to show that FAQchat works properly; it is not a search engine, but it could be a tool to access web pages, and giving answers from FAQ databases.

6. Results and samples of chatting

As a result of the program, FAQchat was enlarged with more than one topic, and a new chatbot was generated “Ask Perl, Python, Linux, SoC” (<http://www.pandorabots.com/pandora/talk?botid=dfe1b84cde365709>), table 2 shows the number of categories generated from each WWW FAQs. In total 13,816 categories were generated. Since the FAQchat does not use any sophisticated NLP, the large number of categories, implies the more opportunity to find a match for a user

request. So a large number of categories increase the ability to find a match.

WWW FAQs	Atomic categories	Default categories
Perl	304	4,024
Linux	170	1,970
Python	189	2,507
SoC	461	4,191

Table 2. AIML categories generated from WWWFAQs

In the previous work, the evaluation of the SoC FAQchat proved that most of answers are related to the questions. Based on that, extra categories are added in this open-ended approach. A sample of chatting with the generated FAQchat is presented in figure 4. A careful look at the sample, we realize that the output depends on the way the question is formatted as appeared with using the word "remove" and "delete" in the third and second question.

Human: what is Perl?

Ask Perl, Python, Linux, SoC: Perl is a high-level programming language with an eclectic heritage written by Larry Wall and a cast of thousands. It derives from the ubiquitous C programming language and to a lesser extent from sed, awk, the Unix shell, and at least a dozen other tools and languages. Perl's process, file, and text manipulation facilities make it particularly well-suited for tasks involving quick prototyping, system utilities, software tools, system management tasks, database access, graphical programming, networking, and World Wide Web programming. These strengths make it especially popular with system administrators and CGI script authors, but mathematicians, geneticists, journalists, and even managers also use Perl. Maybe you should, too. For more information look at: What is Perl?

b. Using the Python FAQ website in English

Human: what is a class ?

Ask Perl, Python, Linux, SoC:

A class is the particular object type created by executing a class statement. Class objects are used as templates to create instance objects, which embody both the data (attributes) and code (methods) specific to a data type.

A class can be based on one or more other classes, called its base class(es). It then inherits the attributes and methods of its base classes. This allows an object model to

be successively refined by inheritance. You might have a generic Mailbox class that provides basic accessor methods for a mailbox, and subclasses such as MboxMailbox, MaildirMailbox, OutlookMailbox that handle various specific mailbox formats.

c. Using the Python FAQ website in Spanish

Human: que es python?

Ask Perl, Python, Linux, SoC: Python es un lenguaje de programación interpretado, interactivo y orientado a objetos. Incorpora módulos, excepciones, tipado dinámico, tipos de datos dinámicos de muy alto nivel, y clases. Python combina un remarkable poder con una sintaxis muy clara. Tiene interfaces a muchas llamadas al sistema y bibliotecas, así como también a varios sistemas de ventanas, y es extensible en C o C++. También es utilizable como un lenguaje de extensión para aplicaciones que necesiten interfaces programables. Finalmente, Python es portable, corre en muchas variantes de Unix, en la Mac, y en PCs bajo MS-DOS, Windows, Windows NT, y OS/2.

d. Using the SoC FAQ website

Human: How can I remove a job from the printer?

Ask Perl, Python, Linux, SoC: If you want to remove a job from the printer queue then find out the job number using the [lpq command](#), and in a UNIX shell type:

```
% lprm -P printername job_no
```

You can only remove jobs that you have sent to the printer yourself - apart from Support that is.

For more information look at:

[How do I remove a job from the printer queue? \(Printing\)](#)

Human: How can I delete a job from a printer?

Ask Perl, Python, Linux, SoC: You may find your answer in the following links:

[How do I check the printer queue? \(Printing\)](#)

[How do I remove a job from the printer queue? \(Printing\)](#)

[How do I check my printer credits? \(Printing\)](#)

[How do I use the printer from a Windows machine?](#)

[\(Other Windows information\)](#)

Figure 4. A sample of chatting with Perl, Linux, Python, and SoC chatbot

6. Conclusion

In this paper, we describe a way to access information using a chatbot, without the need for sophisticated natural language processing or logical inference. FAQs are Frequently-Asked Questions documents, designed to capture the logical ontology of a given domain. Any natural language interface to an FAQ is constrained to reply with the given Answers, so there is no need for deep

analysis or logical inference to map user input questions onto this logical ontology. To test this hypothesis, the FAQ in the School of Computing at the University of Leeds was used to retrain the ALICE chatbot system, producing FAQchat. The replies from FAQchat look like results generated by search engines such as Google. As a result of comparison between FAQchat, Google and AskJeeves, feedback favorable to FAQchat was gained from almost all users, even those who preferred Google. Using the previous evaluation, we extend the knowledge base of the FAQchat in a trend to build an open ended question/answer system, different FAQs from sites were investigated. These FAQs have different annotated format, which needs different normalisation process for each FAQ. This is in turns necessitating the use of standard structure to generate FAQ web pages.

We managed to demonstrate that simple ALICE-style chatbot engine could be used as a tool to access the WWW FAQs. We did not need sophisticated natural language analysis or logical inference; a simple (but large) set of pattern-template matching rules is sufficient.

7. References

- [1] Alice (2002). *A.L.I.C.E AI Foundation*, <http://www.Alicebot.org/>
- [2] Abu Shawar, B., Atwell, E. (2003a). Using dialogue corpora to train a chatbot in: Archer, D, Rayson, P, Wilson, A & McEnery, T (editors) *Proceedings of CL200*, pp.681-690
- [3] Abu Shawar., B., Atwell, E. (2003b). Machine learning from dialogue corpora to generate chatbots. *Expert Update*, vol. 6, pp. 25-30.
- [4] Abu Shawar, B., Atwell, E. (2003c). Using the corpus of Spoken Afrikaans to generate an Afrikaans chatbot. *Southern African Linguistics and Applied Language Studies*. Vol. 21, pp. 283-294.
- [5] Abu Shawar, B., Atwell, E. (2004). An Arabic chatbot giving answers from the Qur'an. In: Bel, B & Marlien, I (editors) *Proceedings of TALN04*. Vol 2, pp. 197-202 ATALA.
- [6] Abu Shawar Bayan, Atwell Eric and Roberts Andy. 2005b. *FAQchat as an information retrieval system*. In: Zygmunt V. (ed.), *Human Language Technologies as a Challenge for Computer Science and Linguistics: Proceedings of the 2nd Language and Technology Conference*, Wydawnictwo Poznanskie, Poznan, pp. 274-278.
- [7] Abu Shawar, B. Atwell, E. (2005). Using corpora in machine-learning chatbot systems. *International Journal of Corpus Linguistics* 10:4, pp. 489-516
- [8] AskJeeves. (2004). [Online]: <http://ask.co.uk/home>
- [9] Boyle, R. (2003). "Understanding search engines". COMP1600: SY11 Introduction to Computer Systems 1. Lecture Notes, School of Computing, University of Leeds. pp65-72.
- [10] Colby, K. (1999). Human-computer conversation in a cognitive therapy program. In Wilks, Y. (eds.) *Machine conversations*. Kluwer, Boston/Dordrecht/London. Pp. 9-19.

- [12] Wallace, R. (2003) *The elements of AIML style*. ALICE AI Foundation.
- [13] Weizenbaum, J. (1966). ELIZA-A computer program for the study of natural language communication between man and machine, *Communications of the ACM*, Vol. 10, No. 8, pp36-45.
- [14] Güzeldere, G. and Franchi, S. (1995). Dialogue with colourful personalities of early ai". In *Constructions of the Mind, SEHR*, volume 4, issue 2. [On line]: <http://www.stanford.edu/group/SHR/4-2/text/toc.html>
- [15] Aitools.org. (2004). [Online]: <http://www.aitools.org/livebots/>
- [16] MUMMI (2004). A Concept for chatbot:"Sanelma" building engaging relationship between the work of art and the exhibition visitor. [Online]: <http://www.mlab.uiah.fi/mummi/sanelma/>
- [17] Cybelle. (2004). [Online]: AgentLand.com
- [18] Internet.com (2007). "Search engine". [Online]: http://www.webopedia.com/TERM/s/search_engine.htm
- [19] Abu Shawar, B.; Atwell, E. (2007) Chatbots: Sind Sie wirklich nützlich? (are they really useful?). *LDV-Forum Journal for Computational Linguistics and Language Technology*, vol. 22, pp. 31-50.