Auto-FAQ-Gen: Automatic Frequently Asked Questions Generation

Fatemeh Raazaghi(✉)

Faculty of Computer Science, University of New Brunswick, Fredericton, NB, Canada
f.razzaghi@unb.ca

Abstract. Using Frequently Asked Questions (FAQs) is a popular way of documenting the list of common questions on particular topics or specific contexts. Most FAQ pages on the Internet are static and can quickly become outdated. We propose to extend the existing work on question answering systems to generating FAQ lists. In conventional Question Answering (QA) systems, users are only allowed to express their queries in a natural language format. A new methodology is proposed to construct the questions by combining of question extraction and question generation methods. The proposed system accepts, extracts, or generates user questions in order to create, maintain, and improve the FAQs quality. In addition to present the basis of QA system, complimentary units will be added to conduct the FAQ list. The research proposed here will contribute to the field of Natural Language Processing, Text Mining, QA, particularly to provide high quality automatic FAQ generation and retrieval.

Keywords: Dynamic frequently asked questions · FAQ · Question Answering · Question Generation · Text mining

1 Introduction

The Question Answering (QA) technology is needed to fulfill the user requirements in the increasing amount of information on the web [3]. The QA systems can be classified into two main categories, open domain and restricted domain. Frequently Asked Question (FAQ) systems have seen been added to those two previously approved research studies [1][2]. FAQs satisfy most notably two goals: first, to provide users with an easy access to browse the key information to solve their problems, and second, to aid the party responsible for frequently answering the same queries from people interested in the same topic. Research in dynamic FAQ generation has the aim of creating lists of questions and answers for FAQ pages automatically or semi-automatically. The scope of the problems that arise in automatic FAQ generation is very broad: from extracting the questions to determining a repetition of a question, formulation of answers, knowledge base creation, and other challenges in automatic content generation.

The proposed Automatic Frequently Asked Question Generation (Auto-FAQ-Gen) system, creates a reliable and dynamic FAQ system. Auto-FAQ-Gen is

D. Barbosa and E. Milios (Eds.): Canadian AI 2015, LNAI 9091, pp. 334–337, 2015.
DOI: 10.1007/978-3-319-18356-5_30

built upon the QA and Question Generation (QG) paradigms to provide an architectural model, a process, system models, and the supporting tools to build FAQs that can adapt effectively in response to different contexts.

2 Challenges

In addition to the difficulties of creating QA and QG systems, developing an automatic FAQ generation system has its own problems:

- *Question Extraction and Question Generation in Question Analysis* What are the effects of QE and QG in Question Analysis? Based on our studies none of the existing QA systems architectures are able to generate questions or extract interrogative sentences [5]. This work will analyze the feasibility of extracting indirect questions or generating latent questions in dynamic FAQ generation.
- *Question Occurrence* How can one determine if a question is asked often? How can one choose relevant questions to add to FAQ lists? There is a lack of methodology to decide on the frequency threshold that qualifies a certain question will appear in an FAQ list or not.
- *Answer Generation* How to generate answers to FAQ questions? Research in the field of automatic QA has focused on factoid (simple) questions for many years [6], and complex types have been mostly ignored by researchers. Complex questions in FAQ lists often seek multiple different types of information simultaneously and do not presuppose that a single answer could meet all of their information needs.
- *Knowledge Base* How to answer Natural Language (NL) questions based on existing knowledge bases? How to create a knowledge base for FAQ generation system?

3 Proposed Work

I intend to take an approach to the problems outlined in Section 2. On the technical side, I will design a framework to automatically build a FAQ page including appropriate questions and related answers. I will elaborate on these components in the following sections:

Question Construction. The quality of an FAQ list depends on predicting the range of questions that a user might ask. Therefore, the questions that will finally be included in the FAQ list for Auto-FAQ-Gen could be generated from three different sources: it can be a question that a user defines explicitly as input to the system (NL format), the output of Question Extraction module, or the result of the Question Generation component. The primary task of this unit is producing a question. Users might use the question mark character in informal language in cases other than questions, a question may be stated in the declaration and indirect form using phrases e.g. "I was wondering..., Could you tell me..., I'd like

to know", or rhetorical questions may not require to be associated with the answer segments [4]. After preprocessing the text, the QE step extracts direct and indirect questions using the Natural Language Parser, rule-based methods, and dictionary. In the QG, given the source text, after the data is cleaned and tokenized, the relevant sentences are passed to the Named Entity (NE) tagger and the Part Of Speech (POS) tagger. The given text can be in a form of a group of simple sentences, complex sentences, paragraphs, or a longer textual entity.

Input and Output Layer. This proposed system receives a written question as input, normally as a set of documents consisting of plain text, or a web page. The final output of the system is a ranked list of FAQs (implicit, explicit, or hidden) and their corresponding answers.

Question Processing. The question processing unit has two components, namely: (1)Question Analysis and (2)Question Interpretation. As the Question Analysis component parses the question string, it determines the expected answer type, extracts named entities and other terms, and creates a concise interpretation of the question consisting of its key phrases and a more specific answer type. Given this information, the Query generator from the Question Interpretation module builds several queries for document retrieval.

Answer Generator. The Answer Generator part contains two modules: Answer Extraction (AE) and Answer Selection (AS). The task of the AE and the AS components are to obtain the desired answer from the best-scored answer candidates and to present the proper formulation back to the user. The search results will pass through a pipeline of filters to produce the final list of ranked answers. A filter can drop unresponsive answers and rearrange the answers according to features such as confidence scores from the Answer Extractor and Ranker component.

Ranker. The Ranker unit contains a Question Ranker, Answer Ranker, and QA set Selector modules. These three modules will employs syntactic, semantic, and machine learning techniques for ranking.

4 Evaluation

This project plans to evaluate various independent components that are defined in Auto-FAQ-Gen independently and also using a theoretical framework to compare the general model with other existing QA systems. For instance, to comprehensive investigate the performance of the Question Generation module a precision and recall based evaluation technique will be used for the TREC-2007 data experiment.

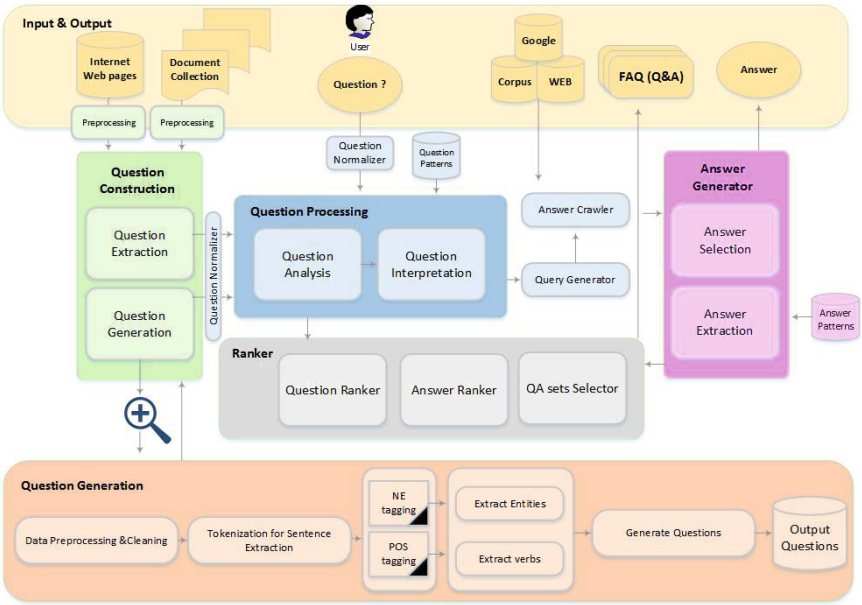

Fig. 1. Architecture of the Auto-FAQ-Gen: Modules and information flow between them

References

1. Hu, W.-C., Yu, D.-F., Jiau, H.C.: A faq finding process in open source project forums. In: 2010 Fifth International Conference on Software Engineering Advances (ICSEA), pp. 259–264. IEEE (2010)
2. Kothari, G., Negi, S., Faruquie, T.A., Chakaravarthy, V.T., Subramaniam, L.V.: Sms based interface for faq retrieval. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, vol. 2, pp. 852–860. Association for Computational Linguistics (2009)
3. Mendes, A.C., Coheur, L.: When the answer comes into question in question-answering: survey and open issues. Natural Language Engineering **19**(1), 1–32 (2013)
4. Shrestha, L., McKeown, K.: Detection of question-answer pairs in email conversations. In: Proceedings of the 20th International Conference on Computational Linguistics, COLING 2004. Association for Computational Linguistics, Stroudsburg (2004)
5. Unger, C., Bühmann, L., Lehmann, J., Ngonga Ngomo, A.-C., Gerber, D., Cimiano, P.: Template-based question answering over rdf data. In: Proceedings of the 21st International Conference on World Wide Web, WWW 2012, pp. 639–648. ACM, New York (2012)
6. Verberne, S., Boves, L., Oostdijk, N., Coppen, P.: Data for question answering: the case of why. In: Proceedings of the 5th edition of the International Conference on Language Resources and Evaluation, LREC 2006, Genoa, Italy (2006)