LOGANSWER IN QUESTION ANSWERING FORUMS

Björn Pelzer
University Koblenz-Landau, Koblenz, Germany
bpelzer@uni-koblenz.de

Ingo Glöckner, Tiansi Dong
University of Hagen, Hagen, Germany
{*ingo.gloeckner*|*tiansi.dong*}*@fernuni-hagen.de*

Keywords: Logic-based Question Answering, Question-Answer Portals, Answer Validation, Answer Bots

Abstract: LogAnswer is a question answering (QA) system for the German language. By providing concise answers to questions of the user, LogAnswer provides more natural access to document collections than conventional search engines do. QA forums provide online venues where human users can ask each other questions and give answers. We describe an ongoing adaptation of LogAnswer to QA forums, aiming at creating a virtual forum user who can respond intelligently and efficiently to human questions. This serves not only as a more accurate evaluation method of our system, but also as a real world use case for automated QA. The basic idea is that the QA system can disburden the human experts from answering routine questions, e.g. questions with known answer in the forum, or questions that can be answered from the Wikipedia. As a result, the users can focus on those questions that really demand human judgement or expertise. In order not to spam users, the QA system needs a good self-assessment of its answer quality. Existing QA techniques, however, are not sufficiently precision-oriented. The need to provide justified answers thus fosters research into logic-oriented QA and novel methods for answer validation.

1 INTRODUCTION

The field of question answering (QA) aims at automatically finding concise answers to arbitrary questions. A QA system communicates with the user in a natural language (NL), its input has the form of properly phrased questions, and the answers are derived from an extensive knowledge base built from a document collection. Conventional search engines with their keyword based search, by contrast, merely produce documents which must be studied further, an approach which is impractical for users with specific questions. Conversely, a QA system only delivers the information that is requested, saving time and allowing a satisfactory result presentation even on compact mobile devices. Its NL interface can be used intuitively, making it suitable for casual users. Drawbacks of QA are the difficulties of dealing with the ambiguities of NL, the need to construct the knowledge base, and the complexity of deriving answers.

LogAnswer (Furbach et al., 2010; Glöckner and Pelzer, 2009) is a QA system for the German language. It works with a knowledge base derived from a local document collection, and the answers are produced using a combination of deep linguistic processing and automated theorem proving. Evaluating QA systems is difficult, as the quality of answers cannot yet be judged automatically. LogAnswer is currently accessible by a web-based interface. To expand its usage and to achieve a better evaluation, we are in the process of adapting LogAnswer to QA forums. Such internet forums normally provide their users with a venue for asking and answering each others' questions (a well-known example is the QA portal `answers.yahoo.com`). By opening up our system to such forums we hope that in the future we can draw from the experiences of a vast number of users in a real-world application of LogAnswer.

In our view, providing technological support for QA forums is an ideal scenario for developing and evaluating QA systems. The potential benefit for users in the forum is just too evident – even if the experts in the forum are only disburdened from answering questions with a known answer in the forum or in the Wikipedia, and even if those asking a question in the forum get an instant, automatically generated answer only for some of their questions. However, the envisioned integration of human QA and automated QA will only succeed if the QA system avoids posting wrong answers, because the users should not be spammed with incorrect responses. This requirement for precision-orientation disqualifies traditional, retrieval-based QA techniques, since these cannot provide the required justified answers. LogAnswer, by contrast, uses logical reasoning for finding and validating answers, thus promising a better self-assessment of answer quality.

The questions found in QA forums are also challenging due to the variety of question types, while many existing QA systems can only handle definition questions and factual questions that ask for a limited number of named-entity types like PERSON, ORGANIZATION etc. In our view, the types of questions that can be tackled by QA techniques and those that should best be answered by human users (like questions asking for an opinion) are complementary. While a QA system can answer questions of the first kind (such as factoid questions) directly, it makes more sense to treat the second kind by techniques for FAQ finding, i.e. by looking up results in a repository of questions with known (human generated) answers. This process of FAQ finding, too, can potentially be improved by incorporating techniques originally developed for question answering.

This paper is divided as follows: In Section 2 we give a short description of LogAnswer. Section 3 provides an overview of QA forums and investigates different adaptation methods for LogAnswer. In the final Section 4 we summarize our results and outline some topics for future work.

2 THE LOGANSWER SYSTEM

LogAnswer is designed as a German language QA system on the web, which can serve as an alternative to conventional search engines. It is accessible by a web interface (`www.loganswer.de`) similar to that of a search engine. The user enters a question into a text box, and LogAnswer then presents the three best answers, which are highlighted in the relevant textual sources to provide a context. The answers are derived from an extensive knowledge base, which has been obtained by automatically translating a snapshot of the German Wikipedia into a semantic network representation in the MultiNet (**Multi**layered Extended Semantic **Net**works) formalism (Helbig, 2006). The WOCADI parser for German (Hartrumpf, 2003) was used for that purpose. Based on its large semantic-based lexicon (Hartrumpf et al., 2003), combined with a robust treatment of unknown words, it achieves a 51.8 percent rate of full parses (80.6 percent including chunk parses) on the Wikipedia. WOCADI uses rule-based and statistical techniques for handling the the various kinds of ambiguities that occur in NL analysis (e.g. prepositonal attachment, resolution of pronouns). For a description of these techniques including a detailed evaluation, see (Hartrumpf, 2003).

The MultNet representations for 29.1 million sentences in the Wikipedia as generated by WOCADI, and an additional 12,000 logical rules and facts constitute the general background knowledge of LogAnswer. Among other things, these additional rules connect the various ways in which a temporal or local specification can be expressed. Examples are shown in (Glöckner, 2007). A large part of the background knowledge is concerned with connecting the meaning representations of verbs and deverbal nouns (e.g. 'read' – 'reader'), or with connecting adjectives and correspoding attributes espressed as nouns (e.g. 'high' – 'height'). Obviously, the results of LogAnswer for a specific application can be improved by adding domain-specific rules that cover paraphrases and other inferences of relevance to the domain.

To make the semantic networks accessible to modern theorem provers, the MultiNet knowledge base has been translated into First-Order Logic (FOL). The expressivity of MultiNet exceeds that of FOL, and while some aspects of MultiNet are lost in the translation, we approximate the expressivity by using logic extensions like equality and arithmetic evaluation. See (Furbach et al., 2010) for a detailed example of the logical translation and subsequent processing.

The complete knowledge base is too large to be handled by an automated theorem prover, in particular when considering that the usage model of LogAnswer requires short response times. Therefore the initial processing steps for a user-provided question serve to narrow down the knowledge base to the fragment relevant to the task at hand. The query is translated into a MultiNet representation, and a machine-learning (ML) ranking technique then uses shallow linguistic criteria (like lexical overlap) to find the text passages most likely to contain an answer. The assessment of these criteria relies on pre-indexed information and can be rapidly computed. The FOL representations (*answer candidates*) of the most promising text passages are then individually tested by the theorem prover E-KRHyper (Pelzer and Wernhard, 2007). In each test the candidate is combined with the background knowledge and the logical query representation as the input for the prover. A successful proof instantiates variables in the FOL question with ground terms representing the answer. Query relaxation techniques increase the likelihood of finding a proof in short time, at the cost of lowering the probability that the answer is relevant for the query, i.e. decreasing the quality of the answer for QA. A second ML phase ranks all proofs according to their quality. The answer terms are collected from the best proofs and translated back into NL answers that are displayed to the user.

Considerable effort has been spent to make the system robust to gaps in the background knowledge and to parsing failure. If a parse of the query fails, then the system is still able to find answer sentences based on robust techniques like predictive annotation (Prager et al., 2000). The robustness enhancing techniques that were developed for LogAnswer are described in (Glöckner and Pelzer, 2008).

3 QUESTION ANSWERING FORUMS

The evaluation of a QA system is difficult for several reasons: Few questions have a single correct answer, as answers can be paraphrased, or the question may be unspecific. A user's acceptance of an answer is very subjective, depending on aspects like intellect and tolerance for malformed answers. Hence it is insufficient to evaluate a QA system using a fixed library of questions with known correct results.

One attempt at a standardized testing is the annual competition of the Cross-Language Evaluation Forum (CLEF: `www.clef-campaign.org`). A set of questions is translated for each participating QA system, and a panel of judges then assesses the answers. Since 2009 the questions refer to documents of the European Parliament, as these texts already exist in multiple translations, but this also means that they have a specialized legal content. While LogAnswer has performed well at CLEF (Glöckner and Pelzer, 2010; Glöckner and Pelzer, 2009), the competition is not representative for a real-world application with actual human users. Moreover, the questions are often closely based on specific documents. Compare for example the question[1] *"Which additives may be used in the manufacture of peeled tomatoes?"* and the answer-containing text passage *"As additives in the manufacture of peeled tomatoes only citric acid (E 330) and calcium chloride (509) may be used."* When question and text passage are very similar, a logical proof using their FOL representations is trivial. The deep reasoning capabilities of a theorem prover are not utilized, instead the bulk of the work is already done once the information retrieval phase has found the passage.

To better evaluate both the real-world applicability and the reasoning aspect of LogAnswer we must therefore look beyond CLEF, and QA forums provide an opportunity for this. We focus on German forums here, but most have English versions as well. In general, the QA forums allow users to ask questions regarding any topic. Users can also give answers to questions that have been asked, and they can browse questions and answers, with new unanswered questions being listed prominently. Many forums allow multiple answers for a question, and the questioner can mark the most helpful answer. *Frag Wikia!* (`frag.wikia.com`) allows only one answer per question, but all users may edit and improve this answer. QA forums are usually financed by online advertising, as for example *Frag Wikia!*, *COSMiQ* (`www.cosmiq.de`) and *WikiAnswers* (`de.answers.com`), or they may be paid directly by the questioners (*JustAnswer*, `www.justanswer.de`). Our preliminary experiments with LogAnswer concentrate on *Frag Wikia!* due to the permissive Creative Commons license for its approximately 80,000 questions. A tighter integration of LogAnswer with any QA forum will require a cooperation with the forum owners, so once our research has progressed we can consider an adaptation to a larger commercial QA forum.

3.1 LogAnswer as a User

The most obvious application of LogAnswer is for the system to assume the role of a virtual forum user who answers the questions that have been posted. At the time of this writing *Frag Wikia!* has a growing backlog of about 5,300 unanswered questions. LogAnswer can reply to questions immediately after they have been posted, so that the questioner does not have to wait an indefinite time for an answer. Alternatively, since forum users do not expect immediate answers, we can allow more time for the logical processing than in the current usage model of LogAnswer, thus enabling deeper reasoning and answers of higher quality, while still ensuring that every new question is answered within a few minutes. Table 1 shows examples of LogAnswer answering forum questions. By covering such encyclopedic questions, LogAnswer disburdens the users in the forum from writing answers to those questions whose answer can easily be found in the Wikipedia. Thus, the users can focus on answering difficult questions that call for specialized

[1] LogAnswer operates on German texts, but for better understanding all examples throughout this paper are in English.

knowledge, human judgement, and advice from experience.

Some questions are well suited to the encyclopedic knowledge of LogAnswer. However, forum users also frequently ask about current events which are not yet covered by the knowledge base of LogAnswer. Many questions are also ambiguous or overly optimistic, for example asking for the phone numbers of celebrities. Nevertheless such questions must be expected in a real-world application.

In a first experiment, we collected a random sample of 200 *Frag Wikia!* questions.[2] We found that 41% of the questions exhibit syntactic mistakes (mostly spelling errors and wrong capitalization), making them difficult to parse automatically. Still, the WOCADI parser used by LogAnswer constructs a logical representation for 81% of the questions. Despite the *Frag Wikia!* policy that questions with an obvious answer in the Wikipedia should be avoided, a manual search revealed that 61% of the questions in our sample find an answer in the German Wikipedia, so that LogAnswer has a chance of answering them. Judging from the results pages produced by LogAnswer that always show three answers, the system achieved a 30% answer rate for these Wikipedia-related questions.

3.2 LogAnswer Compares Questions

As a QA forum is frequented by a multitude of users, the same questions are bound to be asked several times by different people. Most forums try to avoid repeatedly listing such questions as unanswered each time they are posted. Instead an attempt is made to redirect the respective user to an earlier instance of the question, so that old answers can be reused. This requires the forum software to compare the current questions with the older questions in the forum archive. Typically this is done by a search based on keywords from the current question. As a result QA forums may not find a semantically equivalent question if it is a paraphrasing of the current question. For example, *Frag Wikia!* contains the question Q_1: *"What was the name of the first German Chancellor?"*, and asking the same question again will produce this archived question together with an answer. However, rephrasing Q_1 into Q_2: *"Who was the first Chancellor of Germany?"* will not lead to the archived Q_1 and its answer, as Q_2 treated as unique instead.[3] *WikiAnswers* shows a similar behaviour: here the paraphrased question is in the archive and will be found when the identical wording is used. The original question is not found, though. Instead the forum search engine suggests other archived questions containing keywords like *"German"* or *"Chancellor"*, but none of them are relevant for the original question.

LogAnswer can offer an improvement by performing a semantic comparison between questions. This requires a new knowledge base derived from the archived questions. For this purpose the questions must be treated like the Wikipedia text passages for the original knowledge base: the questions are parsed and then translated into their MultiNet and FOL representations. The translation largely corresponds to the way LogAnswer would normally translate a question for question answering, with the exception that constants are used instead of variables. The representations are then indexed by lexical content and by the expected answer types of the questions, allowing efficient access to the question-derived knowledge base. When a user asks a new question, LogAnswer can locate potentially matching archived questions using the same ML filtering phase as for the text passages (see Section 2). The theorem prover then tests the filtered questions for equivalence to the user question. In each test two proofs must be found, one with the user question as a negated conjecture refuted by the archived question and the general background knowledge, and one proof vice versa which refutes the archived question. For example, when attempting to match the aforementioned archived question Q_1 and the user question Q_2, for one proof the prover would operate on the following input[4]:

$$\begin{aligned} Q_1 :& \mathit{pmod(c18, first, chancellor)} \land \mathit{prop(c16, german)} \\ & \land \mathit{sub(c16, c18)} \ldots \\ Q_2 :& \neg \mathit{pmod(X1, first, chancellor)} \lor \neg \mathit{attch(X2, FOCUS)} \\ & \lor \neg \mathit{attr(X2, X3)} \lor \neg \mathit{sub(FOCUS, X1)} \\ & \lor \neg \mathit{sub(X3, name)} \lor \neg \mathit{val(X3, germany)}. \end{aligned}$$

Successful proofs in both directions indicate the semantic equivalence of user question and archived question. Relaxation may be used, possibly weakening the probability of the proofs accurately representing equivalence. In the example LogAnswer finds a refutational proof for Q_2, using both relaxation and a

[2] see `http://www.loganswer.de/resources/icaart2011.xml` for the questions, results of LogAnswer, and annotations.

[3] Note that our rephrased question Q_2 has since been answered by a different user.

[4] The input has been simplified to improve legibility, focussing on those literals which retain a semblance to the NL questions. The original input has over 60 literals and uses more complex symbols to account for different word meanings.

Table 1: Examples for questions from the *Frag Wikia!* QA forum, with answers provided by LogAnswer.

Question	Reply from LogAnswer
How far is the Moon from Earth?	*384,000km*
When was the Berlin Wall built?	*1961*
Who was Ian Fleming?	*British author*

background knowledge axiom which essentially expresses that having a nationality (*"German Chancellor"*) and being of that nation (*"Chancellor of Germany"*) are equivalent. When several archived questions have been tested this way for one user question, then a second ML phase sorts the proofs to determine the archived questions with the highest relevance for the questioner. This process is similar to the second ML phase described in Section 2, except that only a subset of the criteria can be used since a comparison of questions does not produce an answer. LogAnswer then presents the best matching questions from the archive, and the user can examine these and any existing answers.

3.3 Evaluation of LogAnswer by Forum Users

Most QA forums offer an evaluation system which allows questioners to grade the answers they receive or the users who provide them. When LogAnswer is integrated into a QA forum this grading functionality can form the basis for an evaluation of our system. Unfortunately such grading is still in the planning stage in *Frag Wikia!*, which explains our motivation to move on to a more full-featured commercial forum once we have completed our internal adaptations and tests of LogAnswer. For the time being we can utilize the fact that *Frag Wikia!* users may edit and improve answers which they find flawed, thereby indirectly providing us with information about the acceptance of LogAnswer.

4 CONCLUSION AND OUTLOOK

We believe QA forums to be an ideal environment for a system like LogAnswer. They attract a large number of human users who have questions and who are willing to grade the responses they receive, thereby providing a large-scale evaluation of anyone who is able to deliver answers in quantity, including an automated QA system. The questions found on QA forums concern arbitrary topics, and generally they are asked by people who do not know the answers. They might also be ambiguous, malformed, or based on partial or even false information. This means that forum questions require a significant degree of flexibility from a QA system, and a reasoning power which goes beyond pure information retrieval. An adaptation of LogAnswer to QA forums will enable a thorough evaluation of all aspects of our system, allowing us to improve LogAnswer drawing on real-world QA experience. The experimental results give us confidence that this adaptation is feasible.

In our view, combining question answering communities with technologies for automatic question answering is potentially advantageous both for the users in the forum and for QA research. For users asking a question, the most apparent benefit compared to human answerers is the very quick response time of the QA system. But introducing the QA agent also helps the forum experts, who must no longer waste their time with repeated questions or routine questions that can easily be answered from the Wikipedia. In our concrete example forum (`frag.wikia.com`), forum policy discourages asking questions that target at facts from the Wikipedia. Still, we found that 61% of the questions in our sample have an answer in the Wikipedia, so a QA system that uses the Wikipedia as its document collection can indeed be useful. However, such a combination will only be acceptable for the forum users if the involved QA system can realistically judge its answer quality and avoid posting wrong answers. Moreover, the system should be able to integrate automated question answering and FAQ finding from the repository of questions with known answers. This integration is especially important for questions asking for opinions, advice or judgements, where retrieving a human-provided answer is currently the only realistic option.

In the long run, our aim is to develop a virtual and learned internet user who can communicate with real human users, answering questions in several natural languages. Two critical cognitive capabilites for this are: understanding natural languages which, more often than not, contain syntax errors, and finding answers efficiently from a huge knowledge base. As to the first challenge, we are aware that psychologists do not view human errors as malfunctions of the human system, but rather as windows to explore the nature of the human system (Tversky, 1992). Consequently, we do not treat sentences with syntax errors as exceptions. Instead, our language processing mechanism should "understand" sentences as long as humans understand them. To achieve this, we will extend our current method from using MultiNet alone to integrating MultiNet seamlessly with FrameNet, WordNet, and OpenCyc. The basic idea is that the rich background knowledge provided by these resources can often guide linguistic analysis to the intended meaning.

Concerning the second problem, we recall the distinction of implicit knowledge (e.g. intuitions, experiences, procedural knowledge) vs. explicit knowledge (e.g. formal knowledge) (Anderson, 1983). Our current approach in finding answers, which is purely logical, is suited for explicit knowledge but would become awkward for questions on implicit knowledge. Such questions asking for advice, preferences, or judgements are frequent in QA forums, though, so that supporting them seems rewarding. Problems that we will address in this context are: What is the implicit knowledge for a given sentence? How shall we represent it? How can implicit knowledge be connected with our current knowledge representation and reasoning system?

REFERENCES

Anderson, J. R. (1983). *The Architechture of Cognition*. Harvard University Press, Cambridge, MA.

Furbach, U., Glöckner, I., and Pelzer, B. (2010). An application of automated reasoning in natural language question answering. *AI Communications*, 23(2-3):241–265. PAAR Special Issue.

Glöckner, I. (2007). Filtering and fusion of question-answering streams by robust textual inference. In *Proceedings of KRAQ'07*, pages 43–48, Hyderabad, India.

Glöckner, I. and Pelzer, B. (2008). Exploring robustness enhancements for logic-based passage filtering. In *Knowledge Based Intelligent Information and Engineering Systems (Proc. of KES2008, Part I)*, LNAI 5117, pages 606–614. Springer.

Glöckner, I. and Pelzer, B. (2009). The LogAnswer project at CLEF 2009. In *Results of the CLEF 2009 Cross-Language System Evaluation Campaign, Working Notes for the CLEF 2009 Workshop*, Corfu, Greece.

Glöckner, I. and Pelzer, B. (2010). The LogAnswer project at ResPubliQA 2010. In *CLEF 2010 Labs and Workshops, Notebook Papers*. http://clef2010.org/resources/proceedings/clef2010labs_submission_30.pdf.

Hartrumpf, S. (2003). *Hybrid Disambiguation in Natural Language Analysis*. Der Andere Verlag, Osnabrück, Germany.

Hartrumpf, S., Helbig, H., and Osswald, R. (2003). The semantically based computer lexicon HaGenLex. *Traitement automatique des langues*, 44(2):81–105.

Helbig, H. (2006). *Knowledge Representation and the Semantics of Natural Language*. Springer.

Pelzer, B. and Wernhard, C. (2007). System Description: E-KRHyper. In *Automated Deduction - CADE-21, Proceedings*, pages 508–513.

Prager, J., Brown, E., Coden, A., and Radev, D. (2000). Question-answering by predictive annotation. In *SIGIR '00: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 184–191, New York, NY. ACM Press.

Tversky, B. (1992). Distortions in Cognitive Maps. *Geoforum*, 23(2):131–138.