Paragraph retrieval for *why*-question answering

Exploiting discourse structure for intelligent paragraph retrieval for *why-QA*

Suzan Verberne
Department of Linguistics
University of Nijmegen
s.verberne@let.ru.nl

ABSTRACT

Finding answers to *why*-questions involves finding arguments in texts, rather than the noun phrases that are typical targets for factoid questions. Detecting arguments requires detecting specific rhetorical structures and relations. Therefore, we proposed the use of Rhetorical Structure Theory (RST) as a tool for discovering answer to *why*-questions in paragraphs that are likely to contain the answer. We evaluated this method using two sets of *why*-questions: one obtained by elicitation of native speakers and one containing questions that are asked to the online question answering system `answers.com`. Our procedure was able to find answers to about 60% of the *why*-questions. We conclude that some relation types have a high predictive power in answer selection, but we also found that many questions require a full paragraph for an answer. Therefore, we need to shift the research emphasis towards passage retrieval. We propose a three-step method for retrieving passages that are likely to contain the answer to a *why*-question: (1) query creation, (2) document retrieval and (3) paragraph retrieval and ranking. Standard information retrieval models are not suitable for ranking paragraphs as candidate answers. One issue is the small size of the text units that must be ranked. In addition, we need to incorporate information on the presence of RST relations in the language model used for ranking.

General Terms

Question Answering

Keywords

why-questions, paragraph retrieval, discourse structure

1. INTRODUCTION

In the current research project, we aim at developing a system for answering *why*-questions (*why*-QA). Because answers to *why*-questions tend to consist of arguments expressed in complete sentences or sequences of sentences, and because detecting arguments seems to require analysis of relations within and between sentences, we focus on the role that linguistic information and analysis can play in the process of *why*-QA.

Up to now, *why*-questions have largely been ignored by researchers in the field of question answering (QA). One reason for this is that the frequency of *why*-questions posed to QA systems is lower than that of other types of questions such as *who*- and *what*-questions [3]. However, *why*-questions cannot be neglected: as input for a QA system, they comprise about 5 percent of all *wh*-questions [2] and they do have relevance in QA applications [7]. A second reason why this type of question has largely been disregarded until now is that the techniques that have proven to be successful in QA for factoid questions have been demonstrated to be not suitable for questions that expect an explanatory answer instead of a noun phrase [4].

General approaches for QA involve at least four subtasks: (1) question analysis and query creation; (2) retrieval of candidate documents; (3) retrieval and analysis of text fragments; (4) answer generation. In previous research, we focused on two tasks: question analysis and answer extraction from text passages that are likely to contain an answer. We investigated the possibilities of answer extraction for *why*-questions exploiting discourse structure in the source text. In the present paper, we will first discuss the results and the main conclusions that we obtained from our experiments into question analysis and discourse-based answer extraction. One conclusion was that many *why*-questions require a complete paragraph (and occasionally more than one) for an answer. Therefore, we will present our research plans concerning paragraph retrieval for *why*-QA.

2. BACKGROUND AND RELATED WORK

2.1 Question analysis for *why-QA*

In [13] and [14], we focused on question analysis for *why*-questions. From other research reported on in the literature it appears that knowing the answer type helps a QA system in selecting potential answers. In systems for factoid-QA, the answer type is generally deduced directly from the question word (*who*, *when*, *where*, etc.): *who* leads to the answer type *person*; *where* leads to the answer type *place*, etc.

Since determination of the semantic answer type is the most important task of existing question analysis methods [2], we created a question analysis method that was aimed at predicting the semantic answer type. In the work of Moldovan et al. [8], all *why*-questions share the single answer type

SIGIR '07 Amsterdam, the Netherlands

'reason'. However, we believed that it is useful to split this answer type into sub-types, because a more specific answer type can specialize the answer selection algorithm. The idea behind this is that every sub-type has its own lexical and syntactic cues in a source text. Based on the classification of adverbial clauses by Quirk et al. [10], we distinguished the following sub-types of 'reason': 'motivation', 'cause', 'circumstance' and 'purpose'. Of these, 'cause' (52%) and 'motivation' (37%) were by far the most frequent types in a set of *why*-questions pertaining to Reuters and Guardian texts that we elicited from a number of native speakers of English.

We created a syntax-based method for answer type prediction, in which we used the TOSCA system [9] for syntactic analysis and a number of lexical resources like WordNet and VerbNet. With these tools, we extracted a set of feature values from 235 *why*-questions related to 13 texts from Reuters and Guardian. These 235 questions had manually been classified as 'cause' or 'motivation'. The most important features that we used were 'subject agency', 'modality' and 'negation'. We used memory based learning algorithms to classify our questions according to their manually assigned answer type. We evaluated the classification into answer types by using 13-fold cross-validation on the training set. For each round, we tested on the questions for one text, having trained on the questions for the other texts. The best-scoring algorithm (TiMBL) predicted 83.4% of the answer types correctly. Thus, with this syntax-based method, classification is improved by 60% compared to the baseline (classifying all instances as the largest category, viz. 'cause').

2.2 Discourse-based answer extraction

In [15] and [16], we present and discuss an approach to answer extraction for *why*-questions. In approaches to factoid QA, named entity recognition can make a substantial contribution to identifying potential answers. Answers to *why*-questions, on the other hand, cannot be expressed in the form of a noun phrase. Rather, they often span multiple sentences that entertain discourse relations such as 'cause', 'motivation', 'purpose', and 'explanation'. Therefore, we decided to approach the answer extraction problem as a discourse analysis task. In order to investigate to what extent discourse structure enables *why*-QA, we created a system that uses discourse structure for answer extraction.

As a model for discourse annotation, we use Rhetorical Structure Theory (RST), originally developed by Mann and Thompson [6] and adapted by Carlson et al. [1]. In RST, the smallest units of discourse are called 'elementary discourse units' (EDUs). In terms of the RST model, a rhetorical relation typically holds between two EDUs. Two or more related EDUs can be grouped together in a larger span, which in its turn can participate in another RST relation. By grouping and relating spans of text, a hierarchical rhetorical structure of the text is created.

The main reason for using RST in the variant of Carlson et al. is that their rules and guidelines for segmenting EDUs and selecting relations are largely syntax-based, which fits the linguistic perspective of the current research. Moreover, Carlson et al. have created a treebank of manually annotated Wall Street Journal texts with RST structures (the RST Treebank).

Our answer extraction method is based on the idea that the topic of a *why*-question[1] and its answer are siblings in the RST structure of the document, connected by a relation that is relevant for *why*-questions. We implemented an algorithm that (1) indexes all text spans from the source document that participate in a potentially relevant RST relation; (2) matches the input question to each of the text spans in the index; (3) retrieves the sibling for each of the found spans as answer. The result is a list of potential answers, which have been ranked using a probability model that is largely based on lexical overlap. For a more detailed description of our answer extraction method, we refer to [15].

2.2.1 Evaluation data

We evaluated our discourse-based method for answer extraction on two sets of *why*-questions: one obtained by elicitation of native speakers and one containing questions that are submitted to the online QA system `answers.com`.

For the first evaluation collection, we manually selected seven documents from the RST Treebank [1] of 350–550 words each. We created a set of 372 *why*-questions obtained from elicitation of native speakers to these annotated texts. Gathering questions through elicitation entails the risk that subjects might have been tempted to 'invent' *why*-questions that do not address the type of argumentation that one would expect for natural *why*-questions. This may lead to a set of questions that is not completely representative for a user's real information need.

Therefore, we created a second data set, based on the Webclopedia question collection by Hovy et al. [3]. The complete Webclopedia collection consists of 17,000 questions downloaded from `answers.com`, an online domain-independent QA system. 805 questions from the Webclopedia set are *why*-questions—pragmatically defined as questions starting with the word *why*. We randomly selected 400 of these *why*-questions.

For analysis and development purposes, we created a set of answer fragments—varying in size from one sentence to multiple paragraphs—to these 400 questions, manually extracted from Wikipedia. For 54% of the questions, we were able to find the answer in Wikipedia. Of the other 46%, some questions have false question propositions and other questions seem to be either too specific or too trivial for Wikipedia to contain the answer. In a large majority of cases (94%) the length of the answer does not exceed a single paragraph. 61% of the answers is exactly one paragraph, 13% is one sentence, and 20% is longer than one sentence but shorter than a paragraph.

For the purpose of evaluating a method of answer extraction using rhetorical relations, we let three experienced annotators create RST structures for the answer fragments from Wikipedia. For answer fragments shorter than one paragraph, we selected the complete paragraph for annotation. We also added the previous paragraph or the section heading to the fragment if these provided essential information for understanding the paragraph containing the answer. We did not inform the annotators about the purpose of their annotations.

2.2.2 Results and discussion

[1]The topic of a *why*-question is defined as the proposition that is questioned. A *why*-question has the form 'WHY P', in which the proposition P is the topic. [12]

We used both our data collections for evaluating our approach to discourse-based answer extraction. We studied the theoretical upper bound of the contribution of RST to answer extraction by manually analyzing each of the questions for which we have an answer fragment available—and its corresponding RST structure. We manually matched each question topic to a text span in the answer fragment and selected the span's sibling as answer. Following this procedure, we found a satisfactory answer for 58.0% of the question-answer pairs in our set of elicitation data, and for 59.3% of the question-answer pairs in our Webclopedia set. Thus, although the questions in both data collections came from different sources, our answer selection procedure showed highly similar results for both sets.

This analysis shows that the maximum recall that can be achieved using our discourse-based answer extraction approach is around 60%. The remaining 40% of the passages containing candidate answers suffers from one of the following problems: (1) the question topic is not represented by a text span in the answer fragment; (2) the text span representing the question topic does not participate in an RST relation; (3) the correct answer is not the sibling of the span representing the question topic but it is somewhere else in the RST structure.

The RST relations most frequently addressed in our Webclopedia question set are 'elaboration' (31% of the question topics that participate in an RST relation), 'explanation-Argumentative' (16%), 'circumstance' (15%), 'background' (8%) and 'purpose' (5%). Here, we see that the very general relation type 'elaboration' is the most frequently occurring relation type for *why*-questions. However, there is a relatively small proportion of the question topics that participate in an elaboration relation for which this relation leads to a satisfactory answer: 49%. In other words: the predictive power of elaboration relations for *why*-answer retrieval is small. The predictive power for the question topics participating in an explanation-argumentative relation is much larger: for 89% of the question topics that participate in an explanation-argumentative relation, this relation leads to a satisfactory answer. For the question topics participating in a circumstance, background and purpose relation, these relations lead to a satisfactory answer in 77%, 85% and 100% of participating question topics respectively. Thus, we can conclude that the relation types 'explanation-argumentative', 'circumstance', 'background' and 'purpose' are valuable for finding answers to *why*-questions, whereas elaboration relations have low relevance. Furthermore, the predictive power of some types of RST relations confirms the expected importance of answer type determination. If we can predict the answer type from the question, and we know which RST relations represent this answer type ('purpose' relations as defined by Carlson et al., for example, match our definition of 'motivation' as answer type), we can apply the knowledge on the expected answer type for answer selection and ranking.

The analysis described above was done manually. It led to the conclusion that about 60% of the answers to *why*-questions are represented by a relevant RST relation in the source text. In order to investigate the feasibility of the proposed method for implementation in a system for *why*-QA, we implemented our topic matching method in Perl. In the version presented here, our algorithm was optimized for our data collection comprising elicited questions for RST Treebank documents. The procedure for automatically mapping the question topic onto the correct discourse unit in the text mainly uses lexical overlap for finding the discourse unit that is the most similar to the question topic. For the questions related to the RST Treebank documents, 88.7% of the question topics can be identified automatically in the corresponding Wall Street Journal text by this procedure. However, it can only find 41.2% of the discourse units connected to the Webclopedia questions in the Wikipedia documents. This difference is due to the fact that questions elicited from subjects who have been reading a text tend to use the same terms as those that occur in the texts. For the Webclopedia questions such an overlap was not possible, because these questions were formulated completely independently of a specific text. This small lexical overlap for the Webclopedia/Wikipedia collection leads to the problem that in many cases a system relying on lexical overlap cannot match the question topic to the manually chosen text span representing the question topic. This will be the case for all questions posed to a QA system.

We should also note that in realistic applications of *why*-QA using RST, the system will not have access to a manually annotated corpus—it has to deal with automatically annotated data. We assume that automatic RST annotations will be less complete and less precise than the manual annotations are. Consequently, performance must be expected to decline with the use of automatically created annotations. Some work has been done on automatically annotating text with discourse structure. Promising in this direction is the work done Soricut and Marcu [11]. We plan to investigate to what extent we can achieve automatic partial discourse annotations that are specifically equipped to finding answers to *why*-questions.

2.2.3 Conclusion

We found that discourse structure can be useful in solving at least a subset of *why*-questions and that some relation types (the most frequent of which being 'explanation-argumentative', 'circumstance', 'background' and 'purpose') have a predictive power in answer selection. However, our answer extraction approach should be combined with other methods in order to increase recall [16].

In section 2.2.1 we already said that 61% of the answers in our Wikipedia corpus is exactly one paragraph long, 13% is one sentence, and 20% is longer than one sentence but shorter than a paragraph. We studied the answers in the latter two categories (all answers shorter than one paragraph) in order to find out whether the complete paragraph would be a satisfactory answer to the question. We found that the complete paragraph is a satisfactory alternative to 71.8% of the answers shorter than one paragraph. In the other 28.2% of the cases, the paragraph contained too much information on other topics than the core answer. If we add the 61% answers that are exactly one paragraph, we find that for 84.7% of all *why*-questions in our Webclopedia set, a complete paragraph from Wikipedia is a satisfactory answer.

We conclude that paragraph retrieval is a good additive solution to discourse-based answer extraction.

3. PROPOSED RESEARCH

We aim at developing and evaluating an intelligent paragraph retrieval method for *why*-QA. We define a passage as "a fragment of text, longer than a sentence but smaller

than a multi-paragraph document". As stated in section 2.2.1, the length of a large majority of answers (94% of the answers in our Webclopedia/Wikipedia data collection) does not exceed a single paragraph. Therefore, we will consider paragraphs as retrieval units.

We concluded in section 2.2.3 that some types of RST relations have a high predictive power in answer selection. Therefore, we aim at developing a method for paragraph retrieval in which we incorporate knowledge about the presence of relevant RST relations. We formulate the following research question:

> "How can we realize intelligent paragraph retrieval and paragraph ranking for *why*-QA, incorporating knowledge on discourse relations?"

3.1 Method

We explained in section 1 that approaches for QA generally involve query creation, document retrieval, selection and analysis of text fragments, and answer generation. Since we expect that answer generation boils down to presenting paragraphs, we aim at paragraph retrieval for *why*-QA. Thus, we suggest the following approach for answering *why*-questions. (1) question analysis and query creation; (2) document retrieval; (3) paragraph retrieval and ranking.

We did some preliminary research on each of these steps. For the first step, question analysis and query creation, we already mentioned the importance of answer type determination (see section 2.1). So, prediction of the answer type should be part of the question analysis component of our system. As pointed out in section 2.2.2, the system can combine knowledge on the expected answer type and knowledge on the presence of specific types of RST relations to facilitate paragraph selection and ranking. Secondly, we found that lemmatization of query words has a positive effect on the recall of promising passages. Moreover, we did some small experiments to investigate the gain from lexical expansion for retrieval. For our data collection comprising elicited questions for RST Treebank documents, lexical expansion appeared to be of very little help for retrieving discourse units [15]. However, (as mentioned in section 2.2.2) it can probably play a more important role for questions formulated independently from the source text. Our Webclopedia/Wikipedia data set is in that sense representative for questions asked to an online QA system, formulated by persons who do not know the formulations in the documents that (may) contain the answer.

Furthermore, we performed a preliminary experiment into the lexical connection between a *why*-question and the title of the document that contains the answer. We found that the subject/predicate structure of the input question can be helpful in retrieving the answer document. We noticed that the question's grammatical subject often matches the title of the answer document. We made a distinction between semantically poor subjects and semantically rich subjects. A semantically poor subject consists of a pronoun or a noun that is on the top of the WordNet tree or has a direct pointer to a top noun. Examples of semantically poor subjects are *you*, *we*, *people* and *body*. We found that if the subject is semantically rich, it is the subject that leads to the answer document. If the subject is semantically poor, it is the predicate that leads to the answer document. E.g. the answer to the question *Why are flamingos pink?*, which has a semantically rich subject, is in the document on *flamingos*. On the other hand, the answer to the question *Why do we have wax in our ears?*, which has a semantically poor subject, is in the document with title *Ear Wax*. This general rule on the relation between lexical richness of the grammatical subject and the title of the answer document holds for 83% of our Webclopedia questions.

For extracting the subject/predicate structure of the question, we need a linguistic tool that can split the subject and predicate of a *why*-question. Because of the fixed syntactic form of *why*-questions [13], shallow parsing together with fairly simple regular expression matching appeared to suffice for this task. For shallow parsing, we explored the Link Parser [5]. For around 90% of the *why*-questions in our Webclopedia set, we were able to correctly split the subject and the predicate using the Link Parser's output and regular expression matching in Perl.

The considerations above lead us to propose the method for query creation as shown in figure 1 below.

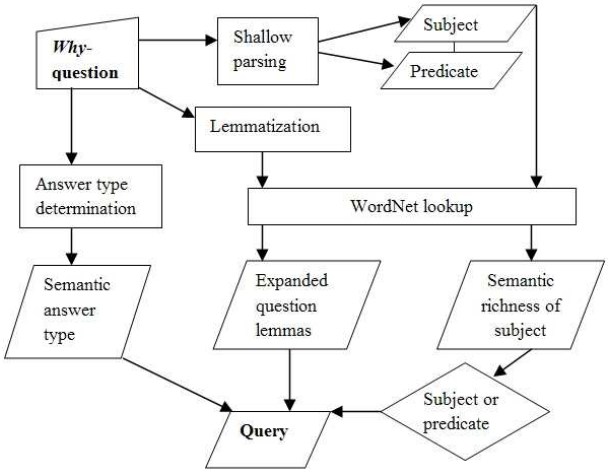

Figure 1: The proposed query creation method

We think that a query existing of the lemmatized and lexically expanded question terms, information on subject and predicate, and the predicted answer type contains sufficient information for performing proper document retrieval.

For ranking the retrieved documents, we aim at incorporating the following variables: lexical overlap between query and document text, lexical overlap between subject or predicate and document title (depending on semantic richness of subject), and a penalty for matching a synonym instead of a literal query term. We will also introduce a threshold for document probability in order to restrict the number of documents returned.

Having a (short) list of documents that possibly contain the answer to the input question, we need an intelligent approach to paragraph retrieval and ranking. In order to be able to retrieve and rank paragraphs, the system has to create an index for each of the paragraphs in a document. Since we want to incorporate knowledge on RST relations, the paragraph index should contain information on the presence of specific RST relations in the paragraph. This information can come from automatic partial discourse annotations, such as the structures created by Soricut and Marcu's Spade tool [11]—which we plan to extend so that it covers the relations that have appeared to be relevant for *why*-QA.

Then, the system has the following information available for paragraph retrieval and ranking: (1) the information from the query; (2) the title of the current document; (3) cues on the locations of potentially relevant RST relations in the text. The presence of some types of RST relations has, as we saw in section 2.2.2, a large predictive power for answer selection. Query information together with the document title can help paragraph retrieval by predicting which part of the query is the most valuable for paragraph retrieval. In the example of *Why are flamingos pink?*, it is the subject *flamingos* that has led to the answer document, and therefore it is the predicate *pink* that is most valuable for paragraph retrieval.

For ranking the paragraphs found, we need to combine the variables 'lexical overlap', 'information on RST relations', and 'document title' into a consistent probability model. The most important variable in general language models for information retrieval is term frequency. We can use lexical overlap as an estimate of term frequency in conventional models. However, general language models are not well geared for retrieving short text units such as paragraphs. This means that we need to develop a language model that is suitable for possibly very short text fragments. Moreover, we aim to incorporate quantifiable information on the presence of RST relations in the language model.

4. ISSUES FOR DISCUSSION

In the previous sections, we have described the research that we carried out in developing an approach for *why*-QA and the method that we propose for the next step in development: paragraph retrieval. There are a number of open issues that we would like to discuss at the Doctoral Consortium meeting:

- **Retrieval of short text fragments**. General language models for information retrieval are aimed at retrieving complete documents. What kind of language model can we apply for retrieving units as short as a single paragraph?
- **Intelligent paragraph retrieval.** We want our language model for paragraph retrieval not only to retrieve fragments based on lexical overlap, but we want to incorporate knowledge on the presence of RST relations in the text fragment for ranking. How can we incorporate RST relations in a consistent probabilistic language model?

5. ACKNOWLEDGEMENTS

The author would like to thank Daphne Theijssen and Hans van Halteren for their recent work on improving syntax-based question classification, and Lou Boves, Nelleke Oostdijk and Peter-Arno Coppen for their helpful comments on this paper.

6. REFERENCES

[1] L. Carlson, D. Marcu, and M. E. Okurowski. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In J. van Kuppevelt and R. Smith, editors, *Current Directions in Discourse and Dialogue*, pages 85–112. Kluwer Academic Publishers, 2003.

[2] E. Hovy, L. Gerber, U. Hermjakob, C.-J. Lin, and D. Ravichandran. Toward semantics-based answer pinpointing. In *Proceedings of the DARPA Human Language Technology Conference (HLT)*, San Diego, CA, 2001.

[3] E. Hovy, U. Hermjakob, and D. Ravichandran. A question/answer typology with surface text patterns. In *Proceedings of the Human Language Technology conference (HLT)*, San Diego, CA, 2002.

[4] J. Kupiec. Murax: Finding and organizing answers from text search. *Natural Language Information Retrieval*, pages 311–332, 1999.

[5] J. Lafferty, D. Sleator, and D. Temperley. Grammatical trigrams: A probabilistic model of link grammar. Technical report, Pittsburgh, PA, USA, 1992.

[6] W. Mann and S. Thompson. Rhetorical structure theory: Toward a functional theory of text organization. *Text, 8 (3)*, pages 243–281, 1988.

[7] M. Maybury, editor. *Toward a Question Answering Roadmap*, pages 8–11. 2003.

[8] D. Moldovan, S. Harabagiu, R. Pasa, R. Mihalcea, R. Grju, R. Goodrum, and V. Rus. The structure and performance of an open domain question answering system. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL-2000)*, pages 563–570, 2000.

[9] N. Oostdijk. Using the tosca analysis system to analyse a software manual corpus. *Industrial Parsing of Software Manuals*, pages 179–206, 1996.

[10] R. Quirk, S. Greenbaum, G. Leech, and J. Svartvik. *A comprehensive grammar of the English language.* London: Longman, 1985.

[11] R. Soricut and D. Marcu. Sentence level discourse parsing using syntactic and lexical information. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 149–156, 2003.

[12] B. Van Fraassen. The pragmatic theory of explanation. *Theories of Explanation*, pages 135–155, 1988.

[13] S. Verberne, L. Boves, N. Oostdijk, and P. Coppen. 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.

[14] S. Verberne, L. Boves, N. Oostdijk, and P. Coppen. Exploring the use of linguistic analysis for *why*-question answering. In *Proceedings of the 16th meeting of Computational Linguistics in the Netherlands (CLIN 2005), Amsterdam*, pages 33–48, 2006.

[15] S. Verberne, L. Boves, N. Oostdijk, and P. Coppen. Discourse-based answering of *why*-questions. 2007. Accepted for *Traitement Automatique des Langues*, special issue on Computational Approaches to Discourse and Document Processing.

[16] S. Verberne, L. Boves, N. Oostdijk, and P. Coppen. Evaluating discourse-based answer extraction for *why*-question answering. 2007. Submitted for the poster session at SIGIR 2007, to be held in Amsterdam, July 2007.