Finding Explanations in Textual Reports

Author 1, Author 2, Author 3
affiliation
address

Abstract

The paper presents our work on extracting explanations from textual reports which then are used in a Case-Based Reasoning (CBR) system to support the suggested solutions and creating trust in the system.
We describe how explanations are extracted from text and represented in a structured representation, called Text Reasoning Graph. Our hypothesis is that the extracted explanations provide an explicit connection between a description of a problem and its solution. In this work we introduce an automatic evaluation measure for explanations to verify this hypothesis.

Introduction

Explanation is a concepts that is intuitive to understand but hard to define explicitly. They are often used by humans to support decisions and actions. The most common type of explanations are causal, answering the why-question, e.g. "John will be late for dinner because he is stuck in traffic". Other common types of explanations are functional, e.g. "Bats have echolocation for navigation in the dark.", and intentional e.g. "Arthur bought new training shoes because he wants to start training." In general, to determine what is an appropriate explanation it is necessary to consider its goal as recognised by Leake et al. [6] and Sormo et al. [15].

In artificial intelligence (AI) systems, explanations have two roles: *internal* and *external* [1]. The internal ones are used in a reasoning process by the system itself while external explanations are targeted at users of the system to show how the system solves a problem. When the performance task of a system is to explain an anomalous situation, the constructed explanation can play the role of both an internal and an external explanation. Therefore a system capable of explaining a domain phenomena is instrumental for a human trying to understand the same phenomenon.

Schank et al.[11] proposed a case-based approach to explanation. The basic idea of this approach is to store, index and retrieve "explanation patterns" (XPs), which are specific or generalized explanations of events. XPs can be tweaked to adapt them to new situations as elaborated in [12]. This approach was implemented in the SWALE system by [4], which can explain complex real-life problems such as the death of the racehorse Swale, who was successful in his career but died prematurely of the unknown reason. The main weakness of SWALE is that it relies on extensive knowledge engineering to construct domain-specific explanation patterns. Meta-AQUA [3] also deals with story understanding, with a special focus on metacognition, relying on domain-specific top-down generated explanation patterns similar to SWALE.

The approach presented in this paper resolves the knowledge engineering burden by not relying on manually constructed explanation patterns but using free text documents as knowledge resources. The overall assumption of our approach is that text documents may contain explanations that can be extracted and used in a case-based reasoning system. Such documents are incident reports, judicial records or service reports for machines. The main characteristic of the reports should be that they contain reusable knowledge for solving new problems, so we can provide computational support for reusing documented knowledge.

In our previous work we described how to extract case-specific explanations from text and showed their usefulness in the case retrieval and adaptation [14, 13]. In this paper we focus on the role of explanation graphs as a connector between the problem description and the solution parts of a case. Explanations make this connection explicit, allowing to trace the reasoning from the problem description to the solution of a case. A good case explanation should not leave any parts of the solution unexplained. This provides a new evaluation criteria for explanations that we investigate in this paper.

The remainder of this paper is structured as follows. First, we introduce the application domain of safety investigation reports. Those reports are used throughout the paper describing our approach. The following two sections describe how cases are built from raw text. Then we present our evaluation introducing a novel evaluation measure for case explanations. Finally we sum up our work and outline future directions.

Application Domain

In our work we use investigation reports from Transportation Safety Board of Canada (TSBC). These reports document the analysis of transportation incidents including aviation, marine, railroad and pipeline incidents. These reports are produced as a result of the incident investigations conducted by the domain experts. The reports are reviewed by the authorities to ensure the quality of the investigation and are published for public access. Currently the repository contains 922 aviation, 375 marine, 298 and 29 pipeline incident reports. Each report is a 5-10 pages long containing summary, factual information, analysis and conclusion sections. Since our goal is to automate the explanation extraction from the analysis task, we focus on the knowledge contained in the analysis section of a report. This is an example of the analysis section obtained from the aviation investigation report (identification number A06Q0091):

> The oil that burned away did not return to the tank and, after a short time, the oil level became very low, causing the engine oil pump to cavitate and the engine oil pressure to fluctuate. Furthermore, since the oil did not return to the tank, the oil temperature did not change, or at least not significantly, and the pilot falsely deduced that the engine oil pressure gauge was displaying an incorrect indication.

Generally speaking, the analysis is essential for solving complex problems such as diagnosing a patient, investigating an accident or predicting the outcome of a legal case. It is a non-trivial process even for human experts and this is the reason why we investigate methods to support human analysts in this process. The type of the analysis we are aiming for is closely related to root cause analysis (RCA) used by human analysts to answer why a certain problem occurred in the first place [10]. A problem is characterised by an undesired outcome such as a failure, accident, defect or a dangerous situation etc. Causes are the events or conditions that lead to the undesired outcome, removal of which would prevent the occurrence of the outcome. RCA goes beyond the proximate causes that immediately and directly precede the outcome and aims to find the root causes that create the proximate causes which in turn ultimately (and possibly indirectly) leads to the outcome. The root causes are of vital importance for the prevention of the same problems to occur in the future.

Mapping Reports to Cases

While studying the investigation reports we saw that all of them have a similar organization where there are certain sections that can be defined either as summary, analysis and conclusion parts. The summary part provides a brief description of the incident and is mapped to the problem description of a case as shown in Figure 1. The analysis part explains why the incident happened, identifying its root causes and contributing factors. It is worth noting that the explanation is not just a label, it is a coherent chain of states or events that links symptoms to the plausible causes. These causes are enumerated in the conclusion part of the report. Both analysis and conclusion parts are considered as case solutions.

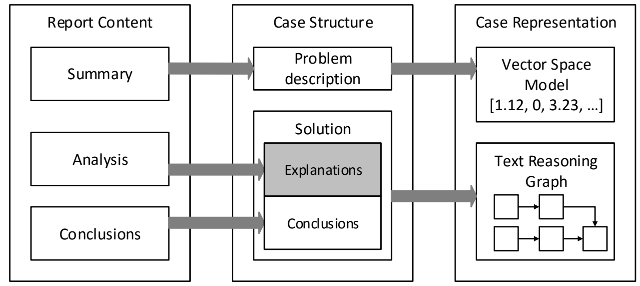

Figure 1: Mapping of textual reports to cases.

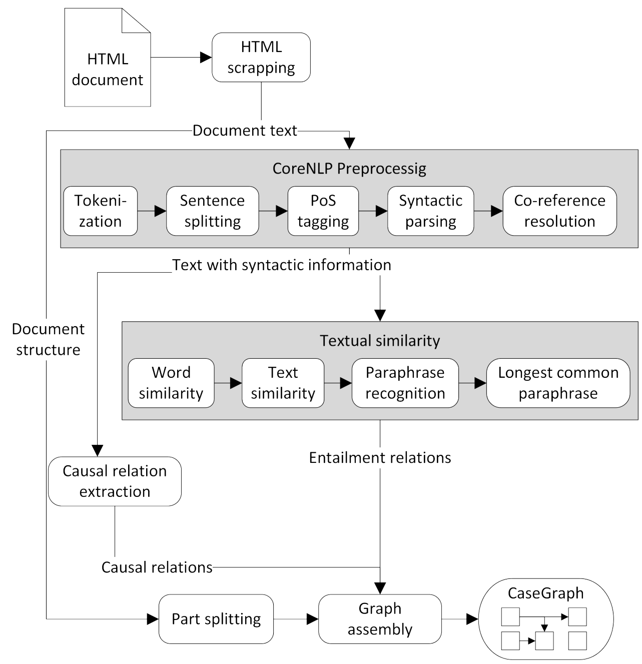

Figure 2: Pipeline for automatic acquisition of explanations from text: extraction and syntactic and semantic analysis of text in order to build CaseGraphs.

Acquisition of Cases from Reports

The extraction process makes extensive use of natural language processing components that are integrated into a pipeline shown in figure 2. As shown in Figure 1 the problem description part of a case is represented as a Vector Space Model, which is obtained by using standard information retrieval methods. The solution of a case is the explanatory graph represented with the Text Reasoning Graph (TRG) representation. It serves as a comprehensive explanation that promotes understanding of why and how the incident under consideration happened and revealed. Figure 3 shows an example of a TRG extracted from an aviation incident report. TRG have the expressive power to represent the chain of reasoning underlying the analysis and enable an efficient mechanism for reuse.

The process of extracting problem solutions from text can be divided into two phases: information extraction and graph

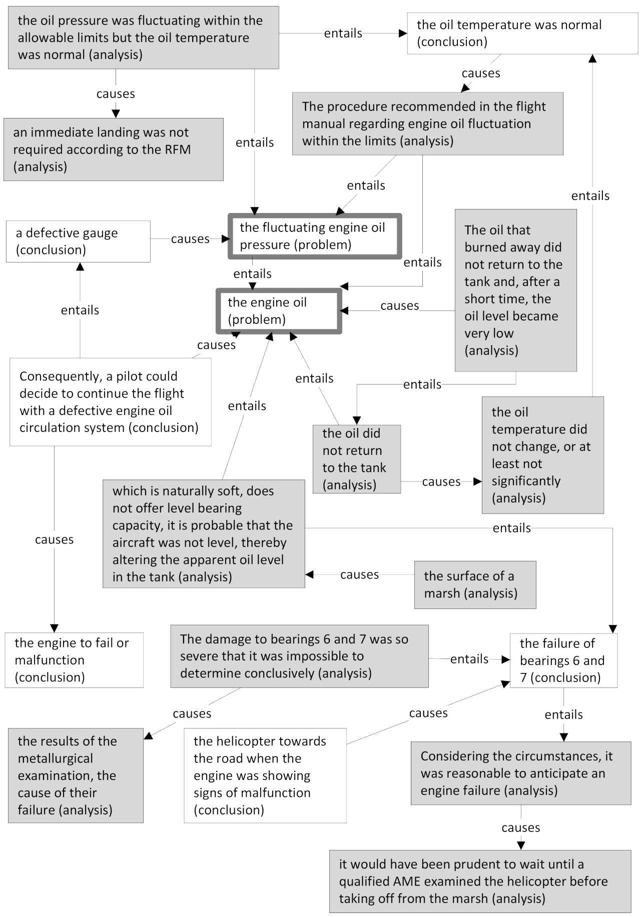

Figure 3: The text reasoning graph containing phrases and relations between them. The problem phrases are framed bold, the analysis phrases are in grey boxes and the conclusion in white boxes.

assembly In the information extraction phase, a report in the HTML format is converted into a structured text annotated with syntactic and semantic information. This information is then used in the graph assembly phase to generate a TRG. The following steps are included in the information extraction phase:

1. Parse the HTML of the report and extract text and sections.
2. Split the report into summary, analysis and conclusion parts based on the section titles, e.g. a section with the title containing the words "findings" or "causes" is assigned to the conclusion part. Similar lexical patterns were constructed for each part.
3. Process the report with the CoreNLP pipeline [7] that includes tokenization, sentence splitting, part-of-speech tagging, syntactic parsing and co-reference resolution (see Figure 2).
4. Extract causal relations from text of the report using patterns proposed by Khoo [5]. These patterns are constructed around causal expressions such "because of" or "as a result of" and incorporate phrasal categories e.g. NP - noun phrase, VP - verb phrase. For example the pattern "because of [CAUSE:NP],[EFFECT]" would match the sentence "Because of the durability of the coverings, it would be extremely difficult for a survivor with hand injuries to open the survival kit." and extract "the durability of the coverings" as the cause and "it would be extremely difficult for a survivor with hand injuries to open the survival kit" as the effect.

The extracted information is then applied to construct the TRG following these steps:

1. Causal relations are collected in one graph with arguments as nodes and relations as edges.
2. Nodes that are not arguments of the same causal relation are connected by textual entailment relations [2] to make the graph more connected. These relations are identified based on the textual similarity measure that uses WordNet [8] to recognise words with similar meaning.
3. Nodes that are paraphrases of each other are merged into one node to remove redundant nodes with a very similar meaning. The paraphrases are detected using the same textual similarity measure as for the textual entailment recognition.
4. Nodes with low informativeness are removed from the graph as described in the section below, e.g. the phrase "after a short time" does not carry much information by itself and is considered uninformative. Formally the informativeness is measured by computing the inverse document frequency (IDF) of the words [9], which is often used in information retrieval for feature weighting.

The details of each step are described in our previous work [14].

Evaluation

In our previous work we evaluated the use of textual explanations in retrieval and adaptation tasks, showing improvements compared to information retrieval baselines [14, 14]. In this work we evaluate the explanations as a connector between problem and solution parts within a case. A good explanation should be able to link anomalies in the problem description to solutions. Based on this criteria we propose a novel evaluation measure called *explanation competence* that shows whether an explanation was able to explain the solution in terms of the problem description. Ideally the whole solution should be explained.

For evaluation we use three incident report data sets from the Transportation Board of Canada collection including aviation, marine and rail incident reports. Each incident report data set contains a conclusion part which enumerates causes and contributing factors for the incident. We consider each cause and factor as a separate conclusion and define our evaluation as the ratio between explained conclusions and the total number of the conclusions in the case:

$$competence(case) = \frac{|conclusion \in explanation|}{|conclusions|} \quad (1)$$

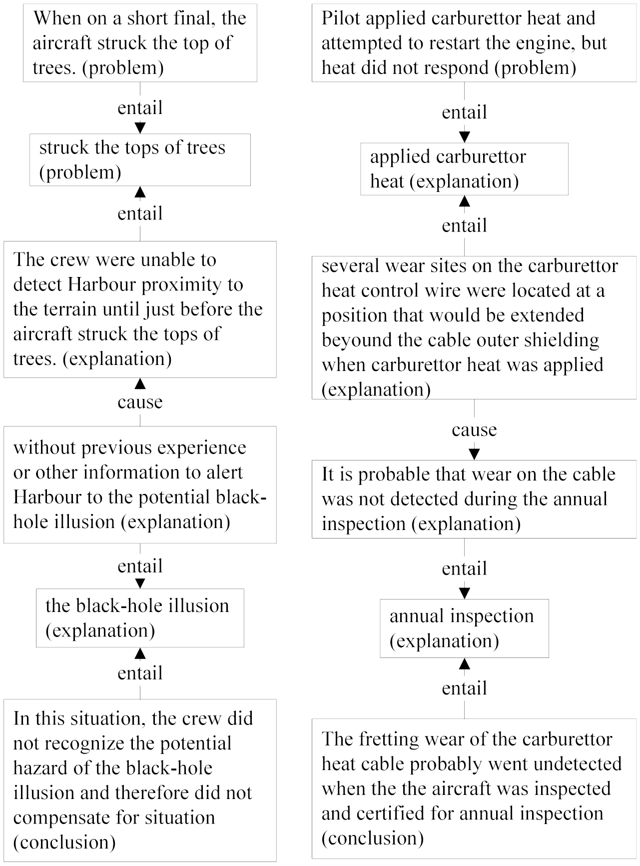

Figure 4: Example of explanations connecting problem (top nodes) to conclusions (bottom nodes)

Dataset	Number of cases	Competence
Aviation	885	0.55
Marine	378	0.53
Rail	275	0.64

Table 1: Mean competence of explanations

A conclusion is considered explained if it is connected to the problem description through the explanation chain. Figure 4 shows examples of explanation chains that connect conclusions to pieces of information from the problem description. It is reasonable to assume that all the conclusions in the report are explained. Otherwise the report would not be approved as lacking rigor. It means that a perfect explanation extraction pipeline would achieve a competence score of 1. However, as shown in table 1, the competence scores obtained by our system are significantly lower, indicating that our explanation acquisition pipeline has room for improvement. The less than ideal scores can be attributed to errors in the NLP components. The causal relation extraction component is particularly important because causal relations are the core of the text reasoning graph representation. Missing relations lead to the broken links between the problem description and the conclusions. We conducted the evaluation of the causal relation extraction component on manually annotated 18 aviation reports with 410 relations. The obtained results, 64.39% precision, 51.85% recall and 56.61% f-score, confirming that the causal relation component

Conclusion and Outlook

In this paper we described how explanations contained in text can be extracted to a structured representation. For evaluation we proposed the explanation competence measure that evaluates explanations based on the ability of explanations to connect problem description to conclusions in the solution. The results of the evaluation indicate that our explanation extraction pipeline has a lot of room for improvement, in particular the causal relation extraction component.

We believe that automatically detecting and providing explanations in CBR will help the user understanding complex problems as well as ease the understanding of a suggested solution. The approach investigated in our research can be used in other domain where documents describing problem solving experiences are available. Our explanation extraction pipeline mostly relies on off-the-shelf NLP components that are available for many big natural languages such as English and German but might no be available for smaller languages.

References

[1] Agnar Aamodt. "A knowledge-intensive, integrated approach to problem solving and sustained learning". In: *Knowledge Engineering and Image Processing Group. University of Trondheim* (1991), pp. 27–85.

[2] Ion Androutsopoulos and Prodromos Malakasiotis. "A Survey of Paraphrasing and Textual Entailment Methods". In: *Journal of Artificial Intelligence Research* 38 (2010), pp. 135–187.

[3] Michael T. Cox and Ashwin Ram. "Introspective Multistrategy Learning: On the Construction of Learning Strategies". In: *Artif. Intell.* 112.1-2 (1999), pp. 1–55.

[4] Alex Kass, David Leake, and Chris Owens. "SWALE: A program that explains". In: *Explanation patterns: Understanding mechanically and creatively* (1986), pp. 232–254.

[5] Christopher Soo-Guan Khoo. "Automatic identification of causal relations in text and their use for improving precision in information retrieval". PhD thesis. The University of Arizona, 1995.

[6] David B Leake. "Abduction, experience, and goals: A model of everyday abductive explanation". In: *Journal of Experimental & Theoretical Artificial Intelligence* 7.4 (1995), pp. 407–428.

[7] Christopher D Manning et al. "The Stanford CoreNLP natural language processing toolkit". In: *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. 2014, pp. 55–60.

[8] George A Miller. "WordNet: a lexical database for English". In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[9] SE Robertson. "Term specificity [letter to the editor]". In: *Journal of Documentation* 28 (1972), pp. 164–165.

[10] James J Rooney and Lee N Vanden Heuvel. "Root cause analysis for beginners". In: *Quality progress* 37.7 (2004), pp. 45–56.

[11] Roger C Schank. "Explanation: A first pass". In: *Experience, memory, and reasoning* (1986), pp. 139–165.

[12] Roger C Schank and David B Leake. "Creativity and learning in a case-based explainer". In: *Artificial Intelligence* 40.1 (1989), pp. 353–385.

[13] Gleb Sizov, Pinar Öztürk, and Agnar Aamodt. "Evidence-Driven Retrieval in Textual CBR: Bridging the Gap Between Retrieval and Reuse." In: *ICCBR-2015, Case-Based Reasoning Research and Development*. Ed. by Eyke Hüllermeier and Mirjam Minor. Vol. 9343. LNCS. Springer, 2015, pp. 351–365.

[14] Gleb Sizov, Pinar Öztürk, and Jozef Styrak. "Acquisition and Reuse of Reasoning Knowledge from Textual Cases for Automated Analysis". In: *ICCBR-2014, Case-Based Reasoning Research and Development*. Ed. by Luc Lamontagne and Enric Plaza. Vol. 8765. LNCS. Springer, 2014, pp. 465–479.

[15] Frode Sørmo, Jörg Cassens, and Agnar Aamodt. "Explanation in case-based reasoning–perspectives and goals". In: *Artificial Intelligence Review* 24.2 (2005), pp. 109–143.