A Conceptual Framework for Semantic Case-based Safety Analysis

Olawande Daramola, Tor Stålhane
Department of Computer and
Information Science
Norwegian University of Science and
Technology, Norway
{wande, stalhane}@idi.ntnu.no

Thomas Moser, Stefan Biffl
Christian Doppler Laboratory for
Software Engineering Integration for
Flexible Automation Systems
Vienna University of Technology, Austria
{thomas.moser, stefan.biffl}@tuwien.ac.at

Abstract

Hazard and Operability (HAZOP) Analysis and Failure Mode and Effect Analysis (FMEA) are among the most widely used safety analysis procedures in the development of safety-critical and embedded systems. These analyses are generally perceived as complex and time-consuming, hindering an effective reuse of previous results or experiences. In this paper we present a conceptual semantic case-based framework for safety analysis, which facilitates the reuse of previous HAZOP and FMEA experiences in order to reduce the time and effort associated with these analyses. We present the core technologies of the conceptual framework and evaluated a prototype of the framework, KROSA, in an experiment with domain experts at ABB Norway. Initial results confirm the viability of the conceptual framework for industrial application.

Key words: *Safety analysis, HAZOP, FMEA, ontology, requirements, case-based reasoning, natural language processing.*

1. Introduction

Safety analysis is an important aspect of embedded systems development, which is concerned with the identification and mitigation of potential hazards, faults, and failure modes of a system. Prominent safety analysis procedures, such as Hazard and Operability (HAZOP) Study and Failure Mode and Effect Analysis (FMEA), are iterative procedures that are time consuming, costly, and typically require a lot of human involvement [1, 2, 3].

HAZOP is used to study potential hazards and operability problems by investigating the effects of deviation from design intent in order to mitigate the occurrence of adverse consequences. The HAZOP process is inherently subjective relying mostly on professional experience, observation, judgement, and creativity of the involved experts. Crucial challenges of HAZOP are how to H1) reduce the level of subjectivity; H2) reduce the human effort; H3) enable the reuse of valuable knowledge gained in previous HAZOP analyses; and H4) facilitate

the sharing and transfer of HAZOP experiences among HAZOP teams [3, 4].

FMEA is a safety analysis procedure used to identify the consequences of failure modes of system components and their associated functions early in the system development life cycle, and to mitigate the effect of these failures. According to Dittman et al. [2], major challenges of FMEA can be classified as:

F1) *knowledge supply problem*: useful information for FMEA is hardly available in structured format with clear semantics such as databases but rather in textual form;

F2) *knowledge processing problem*: previous experiences and facts are hard to utilize;

F3) *complexity problem:* a huge investment of time and human resources is required;

F4) *integration problems*: it is difficult to connect expertise from different aspects of an enterprise into a combined body of knowledge of quality engineering for personnel development;

F5) *searching problem*: it is difficult to provide appropriate search criteria to query FMEA knowledge; and

F6) *update problem*: it is difficult to keep the knowledge in the knowledge base updated especially to avoid redundant work.

These problems motivate the notion of a semantic framework based on case-based reasoning for safety analysis. We believe that an approach that is based on the creation of semantic models of domain-relevant facts using an ontology, the semantic analysis and processing of textual information using natural language processing (NLP), and the reuse of previous experiences in new problem scenarios using case-based reasoning (CBR) has potential to address these major challenges of HAZOP and FMEA. Hence, our conceptual framework leverages three core technologies.

First, CBR - a model of instance-based learning that enables the reuse of previous gained knowledge in resolving a new case [5]. The CBR paradigm inherently supports knowledge acquisition (F1 - *knowledge supply problem*), knowledge retrieval (F5 - *searching problem*), knowledge reuse (F2 - *knowledge processing problem*) and knowledge retention (F6 - *knowledge update problem*). Also, CBR could help a great deal in addressing the problems of HAZOP that are outlined as H2, H3, and H4.

Second, ontology [2] - a semantic representation of the shared formal conceptualization of the facts of a domain that provides a platform for the standardization of terms and vocabulary in the domain (F6 - *update problem*), and enable interoperable transmission of knowledge within the domain (F4 - *integration problem*). Ontology also helps to address problems of HAZOP outlined as *H3* and *H4*.

Third, natural language processing (NLP) [6] - enables the extraction of useful knowledge from natural language documents, and processing of natural language artefacts (F2 - *knowledge processing problem, and problems of HAZOP outlined as H1, and H2*). Consequently a combination of CBR, Ontology, and NLP paradigms looks promising for addressing the major challenges of HAZOP and FMEA.

The aim of the framework is to provide credible tool support for safety analysis processes in order to facilitate early identification of potential system hazards, and enable the interoperable reuse of knowledge for both HAZOP and FMEA activities. It is believed that the reuse of knowledge from previous HAZOP and FMEA experiences has the potential to facilitate reduction in the time and effort associated with these safety analysis procedures. In this work a novel ontology-based framework that enables the reuse of experience for safety analysis right from the requirements engineering phase of system development is presented.

This paper reports on the core technologies of the conceptual framework, we conducted an evaluation of the usability and usefulness of a prototype of the framework, KROSA (Knowledge Reuse-Oriented Safety Analysis), using domain experts at ABB Norway. Initial results confirm the viability of the conceptual framework for industrial application.

The remainder of this paper is organized as follows. Section 2 presents related work, while section 3 gives an overview of the architecture of the conceptual framework. In section 4, a report from the preliminary evaluation of the tool prototype of the conceptual framework is presented. Section 5 concludes the paper and identifies future work.

2. Related Work

Until recently, most of the attempts for automating HAZOP were based on rule-based expert systems [1, 7]. The use of CBR for safety analysis is rather new, and few approaches have been reported so far. Sahar et al. [8] report on the development of a HAZOP analysis management system with dynamic visual model aid. The system is based entirely on CBR and lacks ontology support for HAZOP. PHASUITE [4] is a case-based expert system that uses a suite of informally specified ontologies for automated HAZOP analysis. The PHASUITE system enables the modification of existing HAZOP models and creation of new ones based on the knowledge of pre-existing models. It also possesses diagnostic reasoning capability, but with specific appli-

cation to process generic HAZOP in the chemical industry domain. PetroHAZOP [1] is another expert system for automated HAZOP also used in the chemical domain. It leverages the integration of CBR and ontologies for the automation of both process generic and non-process generic HAZOP procedures.

As with HAZOP not much has been reported on case-based FMEA in the literature, although a few authors have alluded to its feasibility [9, 10]. The description of an intelligent system that combines CBR and rule-based reasoning for intelligent reliability centred maintenance (RCM) analysis was described by Cheng et al. [11]; they mention FMEA as one of the phases of RCM.

Also, few instances of ontology-based FMEA approaches have been reported in the literature [2, 10, 12]. Most significantly, Dittman et al. [2] proposed the notion of ontology-based FMEA as a way to address the six problems outlines for FMEA. However, this was mainly conceptual as there was no prototype implementation or case study conducted to validate the approach. Also, the proposed approach does not embrace the aspect of reuse of experience as a way to minimize the complexity problem of FMEA. Additionally, problems related to searching, updating and knowledge processing were not adequately addressed because the proposal overlooks the need for natural language processing capabilities. Our framework for safety analysis that is presented in this paper uses a more holistic approach to tackle the six problems of FMEA as observed in [2], and also the existing challenges of HAZOP.

Currently, the PetroHAZOP [1] and PHASUITE [4] frameworks appear most related to our work since they are based on an integration of CBR and ontologies. However, none of these two systems is equipped with a natural language processing (NLP) capability, they do not support the creation of ontologies, and were not designed with a focus on requirements engineering. Also, the two systems are specific to the chemical industry domain. Hence, as an improvement to these systems, this work introduces a novel semantic case-based framework that enables the reuse of experience for safety analysis right from the requirements engineering phase of system development, which can also be used in different application domains

3. Overview of the Conceptual Framework

The conceptual framework for semantic case-based safety analysis is instantiated as a prototype software tool called KROSA (see Figure 1). KROSA has a modular architecture that builds on the integration of NLP, CBR, and ontology technologies. In the Figure 1, the core system functionalities are depicted as rectangular boxes while the system artifacts are depicted using oval boxes. Directed links between two entities, e.g. *A* and *B* connotes that *A* depends on *B*, while the undirected links represents mutual dependence between entities.

Document pre-processing is a necessary precondition for using the framework. This entails converting source
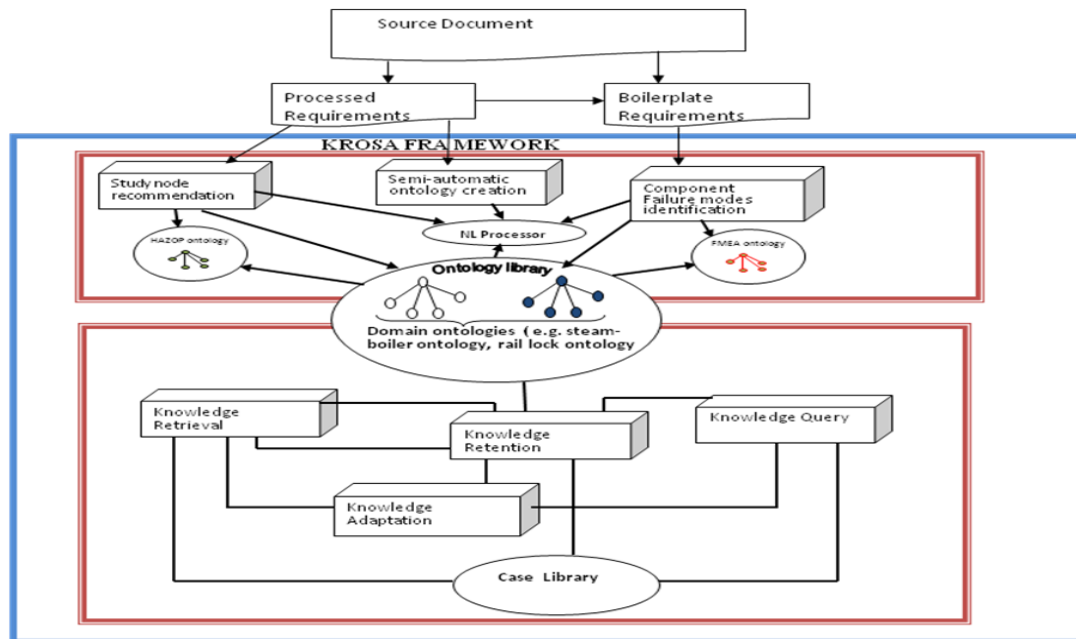
Figure 1. Conceptual Framework for Semantic Case-based Safety Analysis.
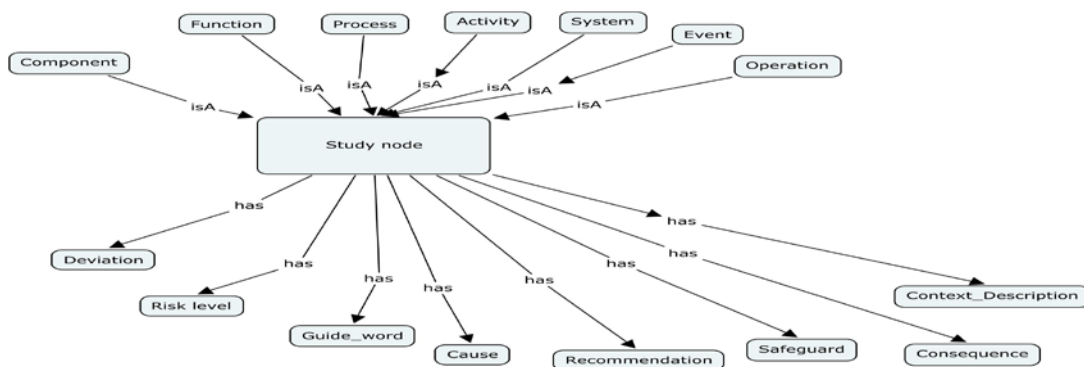

Figure 2. An overview of the Classes in the HAZOP Ontology.

documents into specific file formats (text file or MS Excel), ensuring that sentences have clearly marked boundaries using full stop (.) or being placed on separate lines. Also, information contained in figures, diagrams, and tables must be replaced with equivalent sentences. In the case of requirements documents, the requirements could be transformed into a semi-formal textual form using requirement boilerplates[1]. A boilerplate is a textual template for requirements specification based on prede-fined-patterns in order to standardise the way require-ments are expressed, such that ambiguity that can arise from the use of terms are minimized. More on boiler-plate requirements can be found in [13, 14]. The key components and functionalities supported by the archi-tecture are the following:

HAZOP Ontology: The HAZOP ontology defines the generic concept of a study node, its elements, and their relationships. A study node is a point of observation as pertain to a system or process (e.g. subsystem or com-ponent) that need be examined in order to discover a potential hazard. The description of a study node as specified in the HAZOP ontology includes: types of study node, description, guidewords, deviations, causes, consequences, risk level, safeguards, and recommenda-tion (see Figure 2).

The HAZOP ontology, which we have developed, has three important roles: 1) it is used to identify domain concepts that are contained in source documents that could be study nodes during study node recommenda-tion. Since it describes the types of domain concepts that could be a study node; 2) it is used to validate the struc-ture of the HAZOP information before it is stored in the case library during case retention. A HAZOP study node must be one of the specific types defined in the HAZOP

ontology; 3) the axioms of the HAZOP ontology are used to determine the types of binary associations that can exist between a specific type of study node and other domain concepts. The classes of the HAZOP ontology have specific axioms defined on them that determine the type of relationships they can be involved in. The HAZOP ontology was implemented using the Protégé Ontology Editor environment.

FMEA Ontology: The FMEA ontology which we have developed is a representation of our conceptualisation of FMEA knowledge in a generic form. FMEA knowledge typically consists of unique component name, types, functions, usage contexts, and failure modes. Associated with each failure mode are failure effect, failure cause, importance class, probability of occurrence, probability of detection, risk value number, control method, and containment action. A view of FMEA ontology is shown in Figure 3 using a UML class diagram. The FMEA ontology is used to validate the structure of the FMEA information before it is stored in the case library during case retention. FMEA information must be a valid instantiation of the structure defined in the FMEA ontology. In addition, the FMEA ontology determines the set of possible binary associations that a component can be involved in.

Ontology Library: The ontology library, HAZOP ontology, FMEA ontology, and Case Library are the key semantic infrastructures of the conceptual framework. The ontology library is a repository of domain ontologies. The domain ontologies have either been created for the target domain semi-automatically using the semantic case-based framework environment or existing domain ontologies that have been developed to offer support for safety analysis in specific domains. The domain ontology is used for the identification of valid domain concepts that are contained in requirements documents. Additionally, it supports the process of user data input by ensuring that information that belongs to specific components is automatically retrieved from the domain ontology; and that terms used to describe HAZOP and FMEA information during knowledge (case) retention agree with the established vocabulary of the domain.

NL Processor: This component is used for the processing of natural language texts during the semi-automatic creation of domain ontologies. The NL Processor also operates on requirements documents and boilerplate requirements during the process of automatic recommendation of HAZOP study nodes. The core natural language processing operations involved are:

- *Tokenization*: sentences are split into individual words
- *Parts of speech tagging*: tokens (words) are classified into parts of speech categories such as noun, verb, adjective, pronoun etc.
- *Pronominal Anaphora Resolution*: pronouns that appear as subjects in sentences are associated with noun phrases that precede them. This is essential

for parsing separate sentences that refer to the same requirement.
- *Lexical Parsing*: building a syntax tree that defines the grammatical structure of sentences, with phrases, subjects, objects, and predicates clearly identified.
- *Term Extraction*: extracting subjects and objects, and the predicates that associate them from sentences.
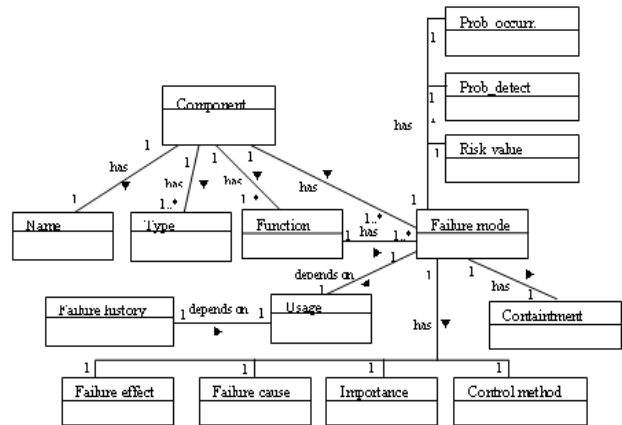

Figure 3. A FMEA Knowledge Model.

The Stanford NLP toolkit[2] for natural language processing was used to implement the NLP operations.

Semi-automatic Ontology Creation: The domain ontology is the semantic infrastructure that contains the vocabulary, and all relevant facts of an application domain. Semi-automatic ontology creation aims to reduce the complexity associated with the creation of usable domain ontology for safety analysis. This involves five activities, namely: pre-processing of document (PD), term extraction (TE), concept identification (CI), ontology creation (OC), and relations mapping (RM). While PD and TE were explained previously, CI entails automatically associating extracted terms with specific concept categories such as: component, system, function, event, failure mode etc. The outputs of CI are subject to revision by the human expert in order to obtain the best results. The CI process employs a knowledge-based approach by using information contained in general-purpose knowledge repositories such as Opencyc ontology[3], Wordnet[4], Wikipedia[5], or the OREDA database[6] (a databank of reliability data on components and equipment used in the offshore oil and gas industry). OC is done automatically after concepts identification. The RM process requires the user to manually link concept pairs together by choosing from a set of suggested possible associations at each instance depending on the category that a concept belongs to. The suggested associa-

tions are based on the descriptions of the different concept categories in the HAZOP and FMEA ontologies. For example, two concepts A and B that are both components would have relations such as *isSubclassof*, *isPartof*, *isEquivalent* suggested as possible associations that can exist between them. It is also possible to create new relations during RM. The RM is an iterative procedure performed for all binary associations considered relevant by the user. The workflow of the semi-automatic ontology creation process is shown in Figure 4.
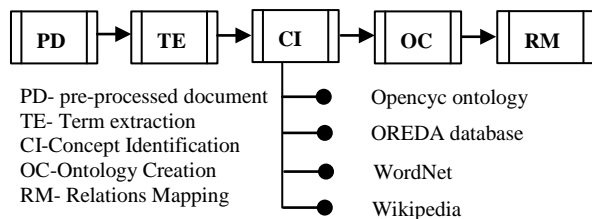

PD- pre-processed document
TE- Term extraction
CI-Concept Identification
OC-Ontology Creation
RM- Relations Mapping

● Opencyc ontology
● OREDA database
● WordNet
● Wikipedia

Figure 4. Semi-automatic domain ontology creation processes.

Study Node Recommendation: The procedure for study node recommendation is based on a heuristic algorithm that is derived from basic knowledge of HAZOP. Study node recommendation creates a credible starting point for early identification of potential hazard, and alleviates the amount of human effort involved. The algorithm searches for potential study nodes in two ways:

- **Requirements level**: A requirement statement is considered as a potential study node if: (1) it contains an *action-entity* pair such as "open valve", "close valve" etc. The *action* and *entity* may not necessarily follow each other in a sentence. The *action* must be an instance of a generic HAZOP action word such as: stop, close, open, send, reset, cut, receive, start or their synonyms, or one of a set of user specified keywords, while *entity* is a valid concept in the domain ontology; (2) it contains terms (domain concepts) that can be associated with any of the predefined study node types e.g. components, system etc., as specified in the HAZOP ontology. Also, the term has failure modes and guidewords defined on it in the domain ontology.
- **Component Level**: A name of a component (term) contained in a requirement statement is considered as a potential study node if: (1) the term is a valid domain concept; (2) there exists at least one axiom (object property relation) that pertains to the term in the domain ontology which indicates that it could be a study node. In other words, it is one of several types of study nodes as defined by the HAZOP ontology. However, a term is overlooked in the selection process if it is: same as, equivalent to, or a subclass of another term that has been selected as a potential study node.

Component Failure Modes Identification: The procedure for component failure mode identification is based on information retrieval from the domain ontology. First terms are extracted from the source requirements document, thereafter, the terms that have been described as components or systems in the domain ontology are identified, and then the failure modes and conditions defined on them are subsequently retrieved. The knowledge contained in the domain ontology is the basis of the identification of component failure modes. The output of the process can be revised by the expert. The Jena Semantic Web Framework[7] was used to implement all ontology management and query functionalities.

CBR Module: The CBR component facilitates the reuse of experience when conducting safety analysis using the conceptual framework. CBR is a pattern-based problem solving paradigm where the features of a new problem and those of previously solved problems are compared in order to adapt solutions from previous cases that are sufficiently similar for solving the new case. In our framework the fundamental concept of the CBR life cycle which is retrieve, reuse, revise, and retain [5, 15] has been adopted. A case model in CBR consists of two parts: the problem part and the solution part. For each new (target) case, similarity algorithms are used to compute the similarity between the problem parts of the new case and the problem parts of all existing relevant cases in the case library to identify suitable candidates for retrieval. Retrieval by the CBR module is performed by displaying a ranked list of cases similar to a target case. Two types of reuse are supported: 1) total reuse - all parts of a case are reused for a new case; 2) partial reuse - only parts of an existing case are used in a target case. Revision is performed by the expert by making the required changes to the selected case to suit the new target case. Retention is done through storage of a specific case into the case library. The case library is implemented as a MySQL[8]database in order to harness its capabilities for effective organization, indexing, storage and retrieval of cases, which ultimately enhances scalability and performance of the CBR module.

3.1 Case Model

The case model is an abstraction of the way HAZOP and FMEA data are modelled as cases in the conceptual framework.

HAZOP Case: An HAZOP case is composed of the following information: name of a study node (unique), context description, type of study node, set of guidewords, set of deviations, set of causes, set of consequences, set of risk level, set of safeguards, and set of recommendations. The three elements of a HAZOP case model that constitute the problem part are context description, type of study node, and set of guidewords. The remaining six elements (set of deviations, set of causes, set of consequences, set of risk level, set of safeguards, and set of recommendations) form the solution part.

---

[7] http://jena.sourceforge.net/

[8] http://www.mysql.com

FMEA Case: An FMEA case is defined in terms of the individual functions associated with a component. Given the set of functions F of a system component C such that F = {f₁, f₂, ... f_m} where $m \in \mathbf{N}$, then a FMEA case can be defined as 11-tuple:

$$FM = <C_{name}, T_{name}, F_{name}, M, S, O, D, P, R, U, H>$$

where

C_{name} - unique name of specific system component C

T_{name} - name of specific type of system component C

F_{name} - name of a specific function f_i of component C

M - set of failure modes of function f_i of C

S - set of importance rating scores for respective failure modes of f_i

O - set of probability of occurrence values for respective failure modes of f_i

D - set of probability of detection values for respective failure modes of f_i

P - represents the set of risk priority number (RPN) for respective failure modes of f_i

R - set of recommended control methods for respective failure nodes of f_i

U - set of containment actions for respective failure nodes of f_i

H - failure history of function f_i of C of type T_{name}

The four elements of the FMEA case model that constitute the problem part are: component name (C_{name}), component type (T_{name}), function name (F_{name}) and the set of Failure modes (M). The remaining elements of FM form the solution part.

3.2 Similarity Algorithms for Case Retrieval

Case retrieval is achieved by using a similarity-based case retrieval algorithm that identifies the most relevant matching cases to a case and ranks them. A number of candidate algorithms can be used for case retrieval depending on the values attributes of data elements of a specific safety analysis process (HAZOP, FMEA). In our case three types of attributes can be identified: object (e.g. a system component, equipment), string (textual descriptions, non-numeric values), and set objects (list). Similarity is determined for each of these cases as follows:

Strings: The similarity of string data is computed based on exact keyword matching of two strings being compared. If the two strings are the same or are synonyms in the domain ontology then a similarity score of 1 is assigned, otherwise it is 0.

Set Objects with string value elements (List): The similarity of two lists with string value elements is determined by computing the degree of set intersection of the two sets. This is given as:

$$Sim(U,V) = \frac{|U \cap V|}{|U|} \qquad (1)$$

where

$$U \cap V = \{x : x \in U \ and \ x \in V\}$$

Object Attributes: Similarity between objects is computed by comparing the semantic description of the two objects as captured in the domain ontology. The path length measure is used to compute object similarity [1, 16]. This is given as:

$$Sim(U,V) = \frac{1}{P} \qquad (2)$$

where

P: is the number of nodes on the shortest path between U and V within the domain ontology hierarchy.

Case Similarity: Finally, the similarity between two cases is computed by using the weighted sum of individual similarity metrics. This is given as [17]:

$$Sim - total = w_1 sim_1 + w_2 sim_2 + ... w_n sim_n \qquad (3)$$

In our implementation the individual similarities have equal weights (i.e., $w_1 = w_2 =... w_n = 1$) because the parameters are considered as equally important.

4. Evaluation

KROSA, a prototype of the conceptual framework for semantic case-based safety analysis is envisioned as a domain-independent CBR platform for ontology-supported HAZOP and FMEA. KROSA is based on the Eclipse plug-in architecture, and its current version implements all features of the conceptual framework except the aspects on FMEA. In this section, we report on a preliminary evaluation of KROSA by three HAZOP experts from ABB Norway, who conducted a usability assessment.

The aim was to assess the viability of the semantic case-based framework for safety analysis by conducting a HAZOP. The focus of the evaluation was to assess 1) the consistency of the outcome of the system as judged by the human experts; 2) the system's overall potential to enable reuse-oriented HAZOP; and 3) its general utility as a support tool for safety analysis. Another objective was to identify areas of possible improvement of the tool.

The inputs to the system needed in order to conduct HAZOP are the pre-processed requirements document and the domain ontology. The ontology could be an existing ontology or an ontology created with KROSA or outside the KROSA environment using any of the standard ontology development tools such as the Protégé ontology Editor. The requirements documented in MS Excel or text file format, and ontologies in OWL and RDF are currently supported.

To conduct the assessment, the ABB experts used three sets of requirements: (1) Rail Lock System (2) Steam Boiler Control system; (3) Adaptive Cruise Control (ACC) System, and three domain ontologies which are: *rail lock system ontology*, *steam boiler ontology* and *ACC ontology*. The steam boiler ontology and ACC

ontology had existed prior to KROSA, having been used to support a previous ontology-based research project in CESAR [13]. The rail lock system ontology was newly created based on information obtained from the system specification obtained from ABB Norway. The three ontologies have the common characteristics that they were developed to be relevant for safety analysis in addition to other uses. Therefore, they contained safety relevant terms, and concepts, and object properties such as: *isComponent*, *isConcept*, *isFailuremode*, *isInterface* etc. Also, the semantic description of components included the definition of generic failure modes like *stuck*, *omission*, and *commission* in many instances.

4.1 The Assessment Procedure

In the expert assessment, the output of the KROSA tool was validated by using an adaptation of the direct method of expert systems evaluation [18, 19]. The direct method for expert system validation is a quantitative assessment that requires the human expert to obtain a copy of the software to be evaluated and to perform a simple benchmark problem on the software. Based on past experience, the human expert answers a set of questions about the software. The questions are quantitative and based on a 0 (completely false) to 5 (very true) numerical scales. From the data obtained, a metric called the "Satisfaction Level" that ranges from 0 (least satisfied user) to 5 (most satisfied user) is computed to rank the system in terms of its likelihood to satisfy a prospective user.

An overview of the content of the questionnaire, the objective of each question and the weight associated with each question (on which the consent of evaluators was sought before the experiment) is shown as follows:

1. *Is sufficient information provided for guidance and orientation of evaluators prior to conducting the experiment?* (Orientation) – *(2)*
2. *Does the KROSA tool reach a conclusion similar to that of a human expert?* (Correctness of Result) – *(2)*
3. *Does the KROSA tool provide reasonable justification for its conclusion?* (Correctness of Result) – *(2)*
4. *Is the KROSA tool accurate in its suggestions of study nodes?* (Accuracy of result) – *(2)*
5. *Is the result complete? Does the user need to do additional work to get a usable result?* (Accuracy of result) - *(2)*
6. *Does the result of the system change if changes are made to the system parameters?* (Sensitivity) – *(1)*
7. *Is the overall usability of the KROSA tool satisfactory?* (Confidence) – *(1)*
8. *Does the KROSA tool give useful conclusions?* (Confidence) – *(2)*
9. *Does the KROSA tool adequately support reuse of knowledge for HAZOP?* (Support for reuse) – *(2)*

10. *Will the KROSA tool improve as data or experience is inserted?* (Support for reuse) – *(1)*
11. *Can limitations of the KROSA tool be detected at this point in time?* (Limitation) – *(1)*
12. *Are there yet many limitations to make the KROSA tool usable?* (Limitation) – *(1)*

For each of the questions an evaluator gives a score between (0-5), based on which a weighted score corresponding to the satisfaction level per evaluator is computed using the metric below:

Result = Σ (weight x score value) / Σ (weight)   (4)

The experts had one full week to interact with the tool, after we had conducted a full day walk-through demonstration session on how to use the tool. The experts were also given a detailed user-manual on the tool.

4.2 Results

Using three safety analysis experts from ABB Norway as evaluators we obtained the mean weighted score of 3.27 (out of 5) as their assessment of the quality of the tool in terms of the three evaluation objectives. The tool obtained its highest mean score ratings in the aspects of: *support for reuse* (4.08), *sensitivity* (3.67), *confidence* (3.25), *and accuracy of result* (3.25), while the lowest mean score ratings were in the aspects of: limitations (3.0) and *correctness of result* (2.7). Each of the experts also returned a detailed report on desired improvements needed to make the tool more usable. Key aspects for improvement as expressed by expert are 1) possibility of providing some form of guidance to users in selection of most appropriate keywords for study node recommendation; 2) need to provide some form of traceability links between cases that have inherited some parts from old cases through reuse. The experts were unanimous in confirming that the tool will be a valuable support for the conduct of HAZOP, with the potential to alleviate the complexity of the process by enabling reuse of experience. They also expressed optimism that benefits of reuse would be derived with regards to FMEA since HAZOP and FMEA are strongly related. Also, relative to the current situation where MS Excel software is the main tool support for safety analysis, they agreed that the existence of a domain ontology and a case library where previous knowledge are stored in a structured format would alleviate some of the existing challenges related to searching, update, and interoperability of knowledge.

5. Conclusion

In this paper a conceptual framework for semantic case-based safety analysis has been presented. In contrast to existing systems, our proposed framework 1) is capable of providing support for several types of HAZOP analyses (process, software, human, or procedure) in different application domains given the existence of relevant domain ontology; 2) embraces a wider

scope by enabling the reuse of experience in HAZOP and FMEA; and 3) provides links to the requirements engineering phase of system development by enabling the early identification of potential system hazards and component failure modes from natural language requirements documents. The evaluation of the framework also confirms its viability for industrial application.

We remain committed to full realization of the complete capability of the conceptual framework for semantic case-based safety analysis. In the future we plan to implement the capability of diagnostic reasoning over potential hazards (component failure modes) in order to facilitate more elaborate support for safety analysis. This is in addition to revisions that will be carried out based on the feedbacks from the evaluation so far conducted. Also, we will look for ways to improve the performance of the NLP algorithms used which are mostly based on a variety of open source tools, and as such our implementations also inherits the limitations of these algorithms.

References

[1] Zhao, J. Cui, L., Zhao, L., Qui T., and Chen, B. (2009): Learning HAZOP Expert System by Case-Based Reasoning and Ontology, *Computers and Chemical Engine*ering, 33(1), pp. 371-378.

[2] Dittman, L., Rademacher, T., and Zelewski, S. (2004): Performing FMEA using Ontologies, In: The 18th international workshop on qualitative reasoning, North-western University, Evanston, Illinois, USA, August 2-4.

[3] Smith, S., Harrison, M. (2005): Measuring Reuse in Hazard Analysis, Reliability Engineering and Systems Safety, 89(1), pp. 93-104.

[4] Zhao, C., Bhushan, M., & Venkatasubramanian, V. (2005): PHASUITE: An automated HAZOP analysis tool for chemical processes Part I: Knowledge Engineering Framework. *Process Safety and Environmental Protection*, *83*(B6), pp. 509–532.

[5] Koldoner, J. (1992): An Introduction to Case-Based Reasoning, *Artificial Intelligence Review*. 6(1), pp. 3-34

[6] Jurafsky, D. and Martin, J.H. (2008): *Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics*. 2nd edition. Prentice-Hall.

[7] Dunjo, J., Fthenakis, V., Vilchez, J. And Arnaldos, J. (2009): Hazard and Operability Analysis. A Literature Review, *Journal of Hazardous Materials,* Doi:10.1016/j.hazmat.2009.08.076

[8] Sahar, B., Ardi, S., Kazuhiko, S., Yoshiomi, M. and Hirotsugu, M. (2010): HAZOP Management System with Dynamic Visual Model Aid, *American Journal of Applied Sciences*, 7(7), pp. 943-948.

[9] Molhanec, M., Mach, P., Mensah, D. (2010): The ontology based FMEA of Lead Free Soldering Process, *In Proceedings of the IEEE 33rd International Spring Seminar on Electronics Technology*, pp. 360-364, Warsar

[10] Atamer, A. (2004) Comparison of FMEA and Field-Experience for a Turbofan Engine with Application to Case-Based Reasoning, *In Proceedings of the IEEE, Aerospace Conference*, pp. 3354-3360.

[11] Cheng, Z. And Jia, X., Gao, P., and Wu, S. (2008): A Framework for Intelligent Reliability Centered Maintenance Analysis, *Reliability Engineering and System Safety*, 93(6), pp. 806-814.

[12] Ebrahimipour, V., Rezaie, K., Shokravi, S. (2010): An Ontology Approach to Support FMEA Studies, *Expert System with Applications*, 37(1), pp. 671-677.

[13] Stålhane T., Omoronyia, I., and Reichenbach, F., (2010), Ontology-guided requirements and safety analysis, Proceedings of Int. Conf. on Safety in Industrial Automation, Tampere, Finland.

[14] Hull E., Jackson, K., Dick, K., Requirements Engineering, Springer, 2004.

[15] Aamodt, A., & Plaza, E. (1994). Case-Based Reasoning: Foundational Issues. Methodological Variations, and System Approaches. *Artificial Intelligence Communications*, *7*(1), pp. 39–59.

[16] Pedersen, T., Pakhomov, S. V. S., Patwardhan, S., & Chute, C. (2007). Measure of Semantic Similarity and Relatedness in Biomedical Domain. *Journal of Biomedical Informatics*, *40*(3), pp. 288–299.

[17] Ferguson, A., Bridge, D., (2000): Generalised Weighting: A Generic Combining Form for Similarity Metrics, *In Proceedings of the 11th Irish Conference on Artificial Intelligence & Cognitive Science (AICS'2000)*, J. Griffith and C. O'Riordan, Eds., pp. 169–179.

[18] Daramola, J.O., Oladipupo, O.O. and Musa, A.G. (2010): A Fuzzy Expert System Tool for Personnel Recruitment, *International Journal of Business Information Systems*, Inderscience, Vol. 6 No.4. [444-462].

[19] Salim M.D, Villavicencio A., Timmerman M.A., "A Method for Evaluating Expert System Shells for Classroom Instruction", *Journal of Industrial Technology,* Vol. 19(1), 2002