Electronics and Electrical Engineering

AIMS Electronics and Electrical Engineering, 4(1): 19–46.
DOI: 10.3934/ElectrEng.2020.1.19
Received: 23 September 2019
Accepted: 12 November 2019
Published: 27 November 2019

http://www.aimspress.com/journal/ElectrEng

Review

Analysis and prediction of railway accident risks using machine learning

Habib Hadj-Mabrouk*

French institute of science and technology for transport, spatial planning, development and networks, Scientific Direction, 14/20 Boulevard Newton, 77447 Marne la Vallée, France

*** Correspondence:** Email: habib.hadj-mabrouk@ifsttar.fr.

Abstract: The harmful consequences of rail accidents, which sometimes lead to loss of life and the destruction of the system and its environment, are at the basis of the implementation of a "experience feedback" (REX) system considered as the essential means to promote the improvement of safety. REX seeks to identify adverse events with a view to highlighting all the causes that contributed to the occurrence of a particular accident and therefore to avoid at least the reproduction of new accidents and similar incidents. Accident and incident investigation reports provide a wealth of informative information for accident prevention. It would be appropriate to exploit these reports in order to extract the relevant information and suggest ways to avoid the reproduction of adverse events. In this context, knowledge of the causes of accidents results mainly from the contribution of lessons learned and experiences gained, whether positive or negative. However, the exploitation of information and the search for lessons from past events is a crucial step in the REX process. This process of analyzing and using data from experience can be facilitated if there are methods and tools available to technical investigators. It seems advisable to consider the use of artificial intelligence (AI) techniques and in particular automatic learning methods in order to understand the origins and circumstances of accidents and therefore propose solutions to avoid the reproduction of similar insecurity events. The fact that the lessons one can learn from a REX depends on the experiences of the situations experienced in the past, constitutes in itself a key argument in favor of machine learning. Thus, the identification of knowledge about rail accidents and incidents and share them among REX actors constitute a process of learning sequences of undesirable events. The approach proposed in this manuscript for the prevention of railway accidents is a hybrid method built around several algorithms and uses several methods of automatic learning: Learning by classification of concepts, Rule-based machine learning (RBML) and Case-based reasoning (CBR).

Keywords: machine learning; case-based reasoning (CBR); rule-based machine learning (RBML); rail safety; accident scenarios; functional safety analysis; software error effect analysis (SEEA)

1. Introduction

This paper describes our contribution to improving the usual safety analysis methods used in the certification of railway transport systems in France. Our approach has been to exploit the historical scenario knowledge base by means of learning with a view to producing knowledge which could provide assistance to experts in their task of evaluating the level of safety of a new system of transport. The purpose is contributed to the generation of new accident scenarios that could help experts to conclude on the safe character of a new rail transport system. After having located our study in the context of feedback experience (Rex) on rail accidents and incidents in order to show the need to resort to artificial intelligence techniques and in particular machine learning, the second paragraph proposes a review of the literature on approaches implemented today to understand the problem of data exploitation. Despite the undeniable interest of artificial intelligence approaches, there is no comprehensive approach to meet all of our research objectives and needs for analysis of railway safety. In order to better situate our contribution, it seems to us essential to specify our research objectives in relation to the safety analysis problem identified during the knowledge acquisition phase with experts in the field. This study is the subject of the third paragraph which describes in detail the purpose of our study as well as the approach taken for the development of two complementary tools to assist in the analysis and assessment of safety. The first tool "Acasya" concerns the Functional Safety Analysis (FSA) and the second tool "Sautrel" relates to the analysis of the safety critical software and more precisely Software Errors and Effects Analysis (SEEA). The "Acasya" tool is based on the joint use of two learning techniques: Learning by classification of concepts and Rule-based machine learning (RBML) to automatically identify, from a base of historical scenarios (experience feedback), the relevant safety rules that are often difficult to extract manually from safety experts. The "Sautrel" tool is based in particular on the case-based reasoning (CBR) in order to find out, from a set of adverse events resulting from feedback on SEEA, the most similar case to a new particular safety problem and finally proposes solutions and recommendations (measures of protection or correction) the most appropriate for dealing with the problem.

2. Experience feedback (REX)

The rail transport system is a complex socio-technical system that requires the many human operators to interact continuously with technology in order to ensure operating that are not only efficient in terms of time, cost and quality, but also which respect a reasonable level of safety vis-à-vis the Man, the system, the environment and their respective interactions [1,2]. The harmful consequences of rail accidents, which sometimes lead to loss of life and the destruction of the system and its environment, are at the basis of the implementation of a “experience feedback” (REX) system considered as the essential means to promote the improvement of safety. Accident and incident investigation reports provide a wealth of informative information for accident prevention. It would be appropriate to exploit these reports in order to extract the relevant information and suggest ways to avoid the reproduction of adverse events. The REX consists, then, in the management of the knowledge coming from a positive and / or negative event making it possible to take the appropriate

decisions in situations of the same nature in the future. The safety management system (SMS) describes how the REX process is developed: 1) identification of adverse events, 2) collection of elements involved, 3) recording of information collected, 4) analysis and implementation highlight all the causes that contributed to the occurrence of the hazardous event and 5) the exploitation of information and the search for lessons to be learned (weaknesses identified, procedures or equipment to evolve, ...). In this context, the knowledge of accidents and incidents results essentially from the contribution of lessons learned and experiences acquired. This process of analyzing and using data from experience can be facilitated if there are methods and tools available to technical investigators. It seems advisable to consider the use of artificial intelligence (AI) techniques and in particular automatic learning methods in order to understand the origins and circumstances of accidents and therefore propose solutions to avoid the reproduction of similar insecurity events. The fact that the lessons one can learn from a REX depends on the experiences of the situations experienced in the past, constitutes in itself a key argument in favor of machine learning. However, the step on data mining is a key element of the REX process. The proposed recommendations (preventive and corrective measures) are closely linked to the quality of the data acquired after an accident or incident. In order to situate our approach in relation to the existing research work, it is necessary to present a detailed study of the methods, techniques and tools coming from the field of artificial intelligence making it possible to contribute to the exploitation of the data involved in the investigation of accidents and incidents.

3. Literature review

The intellectual process by which a human operator evaluates a situation, predicts an event or makes a decision is often difficult to model in the form of reliable and definitive algorithms. This difficulty can be partially overcome by using Artificial Intelligence (AI) techniques. In recent years the considerable development of AI techniques has made it possible to overcome the inadequacy of traditional computing. AI aims to study and simulate human intellectual activities and strives to create machines capable of “intelligent” behavior. Artificial Intelligence aims ambitiously to equip the computer with some of the faculties of the human mind: To learn, to recognize, to reason, etc. [3].

In recent years and in the field of land and air transport, researchers and experts in the field have become increasingly interested in the application of artificial intelligence techniques to solve certain problems of aid the decision, such as the diagnosis of transport equipment, the management of maintenance operations, the analysis of driver behavior, the prediction of the deterioration of transport infrastructure, the planning and forecasting of traffic demand, control of traffic signals, control of air traffic, etc. For example, machine learning has been used for rail maintenance forecasting [4], Expert Systems (fuzzy knowledge based) for rail traffic control [5], deep learning for the detection of lateral defects of the railroad [6] and neural networks for the detection of defects on the surface of rails [7].

In railway transport applications, the Big Data Analytics (BDA) can be of a beneficial contribution in view of the large amounts of data generated by the transport system from sensors installed on the tracks, on the wagons or from the signaling equipment, monitoring and inspection equipment, communication systems, train monitoring systems, etc. A BDA can examine the collected data set in order to obtain useful information to explain for example the potential causes of degradation of the operation, the failure of certain track components and possibly safety equipment.

As an example, we can mention the work on exploiting data relating to operation, maintenance and railway safety [8], decision-making on rail maintenance [9], engineering and the management of railway applications [10], the improvement of call reporting systems [11], the implementation of a predictive approach to the safety and maintenance of personnel [12], and Siemens on the use of Big Data to build the Internet of trains [13]. Contribution of the "Data Mining" approach to retrieving information from accident investigation reports.

In the field of "data mining" and retrieval of relevant information from accident investigation reports, there are several techniques from various fields, such as information retrieval (IR), natural language processing (NLP), information extraction (IE), BDA approach and machine learning. The main goal is to explore plain text in order to extract relevant information for explanatory or decisional purposes. In the field of railway safety, these methods are generally used to extract the presence of informative entities on the causes of accidents, recurrent accidents, to understand the causes of accidents, to find causal relationships from the investigation reports on accidents.

To analyze reports of major railway accidents, Williams et al. [14] use the text mining techniques of probabilistic topic modeling and k-means clustering. The results of this study show that the types of recurring accidents are lane defects, wheel defects, level crossing accidents and switching accidents. Studies also show, through case examples (feedback), how the results of the textual search of stories can improve the understanding of contributing factors to rail accidents. Brown's paper [15] describes the use of text mining (with the combination of other techniques) to automatically discover the characteristics of railway accidents and to gain a better understanding of the factors contributing to these accidents. LI et al. [16] also applied the text mining method to the risk analysis for the safety of urban rail transport in China. The word frequency analysis and cluster analysis identified 15 safety risk factors from 156 accident reports. In the context of text mining, Williams et al. [17] use a comparison between Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA) for railway accident text analysis. Syeda et al. [18] uses a Big Data Analytics (BDA) approach to analyze incident reporting to reduce safety risks in railway projects and operations. Van-Gulijk [19] presents the case for IT transformation and Big Data for managing safety risks on UK railways. To investigate the causal link between causes and safety deficiencies in the rail industry, Kanza et al. [20] uses natural language processing (NLP) and machine learning. The objective is to reveal the presence of informative entities on the causes of rail accidents from the raw texts of accident investigation reports in the United Kingdom (published by the Railway Accident Investigation Directorate: RAIB). Ghomi et al. [21] applies data mining techniques based on association rules and classification algorithms to identify the severity factors of injuries caused by crossing accidents. Exploitation of the accident database shows that train speed, the age of vulnerable road users and sex are the most influential accident factors. In order to improve the identification of accidents, Zhang [22] developed an approach based on machine learning, in order to distinguish secondary accidents. This text mining study shows that the classification model implemented is effective in identifying the keywords that characterize secondary collisions. Heidarysafa [23] uses in-depth learning to analyze railway accident narratives to understand the causes of accidents and their corresponding descriptions in survey reports. The main goal is to help label accidents more accurately. Gibert [24] uses deep learning to inspect the railroad. Osama [25] proposes a machine learning model for near-accident prediction from observed vehicle kinematics data. Chenariyan [26] presents recent applications of machine learning in railway maintenance.

Case-based reasoning (CBR) is attracting more and more attention from researchers and experts in the rail transport sector. This therefore argues for the need to review recent research in this area with a view to providing a comprehensive review of the major recent applications in the

context of rail transport. CBR is a well-established field of research based on artificial intelligence techniques and in particular machine learning, as evidenced by the 27th International Conference on Case-Based Reasoning (ICCBR) held in Stockholm, Sweden from July 10 to 12, 2018. This mode of reasoning, which is based on the notion of similarity, focuses primarily on problem solving based on experience. It is a cognitive process of human reasoning that relies heavily on how people acquire a new skill based on their past habits and experiences. CBR means using and exploiting old experiences to understand, explain, interpret or solve new situations similar to similar past situations. CBRs are increasingly used in industrial applications such as technical diagnostics, medical diagnostics, image processing, law, design, planning, and so on. In the field of transportation, our literature search covered three transport sectors: Air, road and rail. In the field of air transport we can cite, for example, the prediction of accidents and incidents [27]. In the road transport sector, the application of CBR is numerous: Transport planning [28], management of traffic flows [29,30], control of urban intersections to avoid road congestion [31], the analysis of road collisions [32], the improvement of traffic in urban intersections by developing new signaling plans [33], the control of traffic flow at intersections (traffic control systems (TCS)) [34], the diagnosis of the driver's stress level [35], or the modeling of the risk of driver fatigue [36]. Finally, in the rail transport sector, studies include the diagnosis of locomotive failures [37], the recovery of incident reports [38], the prevention of rail operations incidents [39], the command of railway rescue (Emergency Relief Command) [40], analysis of safety risks related to the operation of the metro [41], automatic train conduction to reduce travel time and save fuel consumption [42] and finally the diagnosis of failures of the rail switching system [43].

All of this work clearly shows that AI, BDA and machine learning will likely have an increasing impact on the safety of rail transport.

4. Contributions with respect the state of the art

Despite the undeniable interest of artificial intelligence approaches presented in the previous paragraph, there is no comprehensive approach to meet all of our research objectives and needs for analysis of railway safety. Our research objectives focus on using data from rail accident and incident feedback experience from the design phase of the system and not after the operation of the transportation system. The bibliographic study presented above shows that all the works examined concern the exploitation of historical information relating to technical investigations into accidents after the putting into service of the transport system. It is important to note that there is several feedbacks experience on rail accidents: 1) in the design phase, 2) during the validation and certification of the system and 3) after exploitation and maintenance of the system. Work on the exploitation of accident investigation data is "downstream" from the system's operational phase. They consider that the system is already in service and seek to learn new knowledge from failures in the past to avoid the production of such undesirable events. These studies concern the examination of the causes of accidents, the causal links between the causes and the effects generated, the occurrence of certain events which are contrary to safety, and. But our study is located "upstream" of the commissioning phase of the system and strives to take into account the scenarios of potential accidents from the design phase of the system. Indeed, during the design phase of the project, the system designer proposes in its safety file all the functions and safety equipment. For each safety function (or critical safety task), the designer proposes all the technical means (safety functions, rules, procedures, etc.) that can cover all the potential accident risks identified during the preliminary hazard analysis (PHA). In contrast, during the evaluation phase, certification experts

seek to question the safety measures proposed by the designer. To this end, they are constantly forced to use their experiences and their imaginations to produce new situations that are not foreseen by the designer and that could jeopardize the overall safety of the system. This contradictory approach to safety is usually embodied in the imagination of new scenarios of potential accidents.

The approach proposed in this manuscript for the prevention of railway accidents is a hybrid method built around several algorithms and uses several modes of reasoning: induction, deduction and analogy. In fact, faced with a complex and highly evolving field, such as safety and certification of transport systems, this approach successively calls upon the following methods:

- The acquisition of knowledge to gather knowledge of railway safety and in particular the scenarios of potential accidents,
- Learning by classification of concepts to group accident scenarios into homogeneous classes such as the class relating to train collision or derailment problems.
- Rule-based machine learning (RBML) to automatically identify, from a base of historical scenarios (experience feedback), the relevant safety rules that are often difficult to extract manually from safety experts,
- Knowledge-based system (KBS). Production rules, previously induced by machine learning, are transferred to KBS to form the knowledge base of the safety assessment support tool.
- Case-based reasoning system (CBR). At the previous level, the KBS is used to evaluate safety at the highest level of the safety analysis hierarchy and can deduce a possible risk of accident not taken into account and likely to jeopardize the safety of the system and by therefore the safety of hardware and software equipment. This risk of accident requires the implementation of new prevention or protection measures during the various safety analyzes of hardware and software equipment (low level of the hierarchy). In this context, the CBR makes it possible to look for the most similar cases to this new risk of accident and proposes the appropriate measures.

The bibliographic study carried out on machine learning and in particular on CBR shows the absence of work on the use of CBR in the analysis and evaluation of the safety of critical software used in the rail transport sector. To date and to our knowledge, this is the first work in this area, which is one of the original features of our study.

In order to better situate our contribution, it seems to us essential to specify our research objectives in relation to the safety analysis problem identified during the knowledge acquisition phase with experts in the field.

5. Objectives and motivations of the study

The process of analyzing rail safety can be broken down into several levels: System, automatisms, hardware and software. Each level of safety analysis has one or more safety methods:

- At the system level, the main method is the “Preliminary hazard analysis” (PHA) method. The PHA aims to identify potential accidents related to the transport system and its interfaces in order to evaluate them and propose solutions to remove reduce or control them [44].
- At the level of automatisms, a method known as “Functional safety analysis” (FSA). The FSA aims to justify that the design architecture of the system is safe against potential accidents identified by the PHA and therefore to ensure that all safety provisions are taken into account for cover potential hazards or accidents.

– At a software level, it is a question of carrying out several methods related to Software Safety Analysis (SSA). The SSA is generally based on the “Software Errors and Effects Analysis” (SEEA) method as well as on critical code reads.
– At the hardware level, several safety methods relating to Hardware Safety Analysis (HSA) need to be established. The HSA focuses on electronic boards and interfaces defined of safety.

This analysis implements two types of analysis: inductive and deductive:

– An "inductive" analysis by analysis of failure modes, their effects and their criticality: AFMEC. The AFMEC method is usually completed by the “method of Combining Summarized Failures”, (MCSF) also named Significant Failures Combination Search Method. Coming from the field of aeronautics, the MCSF method was developed jointly by the National Society of Aeronautical and Space Industries (NSASI) and the French Air Ministry Certification Authorities, for the analysis of planes safety Concorde and Airbus. The AFMEC method, which usually highlights simple failures, must be supplemented by studying combinations of failures that result in undesirable (or dangerous) events. Thus, the MCSF method, used in the extension of the AFMEC method, inductively determines such combinations of failures. Generally, it is noted for one or more modes of failure, that the effects (or consequences) on the system are identical. These failure modes are then grouped into fault sets called “Summary Faults” (SF). This method of safety analysis is therefore focused on extracting only the combinations of safety-relevant failures and then presents itself as an extension of the conventional AFMEC method. As part of our approach to analyzing and evaluating rail transport safety, we use this concept of Summary Failure (SF) [45].
– A "deductive" type of analysis by searching for scenarios that run counter to safety and that make it impossible to comply with the safety criteria derived from the “functional safety analysis” (FSA). This deductive analysis usually requires the use of the Cause Tree method.

Our research is part of two complementary safety analyzes: Functional Safety Analysis (FSA) and Software Errors and Effects Analysis (SEEA). The FSA aims to justify that the design architecture of the system is safe against potential accidents identified by the PHA and therefore to ensure that all safety provisions are taken into account for cover potential hazards or accidents. These analyzes provide (low level) safety criteria for the design of the system and the realization of hardware and software safety equipment. They also impose safety criteria related to the sizing, operation and maintenance of the system. FSAs can highlight unsafe scenarios that require specification recovery and system design. SEEA is a safety analysis approach whose purpose is to determine the nature and severity of the consequences of software failures. SEEA guides software validation and maintenance activities by identifying the most critical modules for safety. SEEA makes it possible to estimate the level of effort of validation to be carried out on the various elements of the software and in particular, to guide the readings of code and to better target the tests. This analysis is performed by considering software error assumptions and examining the consequences of these errors on the other modules as well as any system-related failures SEEA finally proposes measures to detect errors and improve the robustness of the software.

Safety experts and certification bodies face several obstacles to improving the safety level of rail transport systems, in particular the difficulty in synthesizing and exploiting historical knowledge of FSA and SEEA (experience feedback) and the willingness to judge the completeness of the proposed analyzes by the manufacturer during the development of a new rail transport system. Thus,

the need to rationalize traditional approaches to safety analysis, to improve the quality of accident risk analyzes and finally to help experts in problem solving and decision-making, has led us to the development of machine learning tools to suggest potential accidents and / or the most appropriate protective or preventive measures to protect against a particular risk. The development of a safety analysis support tool was motivated by various findings revealed by the problem identification and specification phase. These findings guided us towards the development of two tools named "ACASYA" and "SAUTREL". ACASYA, part of the Functional Safety Analysis (FSA), is based on Rule-based machine learning (RBML) to help the generation of potential accident scenarios that could jeopardize the safety of the system. SATREL, which deals with software safety analysis (SEEA), is based on case-based reasoning (CBR). Its purpose is to look for historical situations (source cases) that are more analogous to the new problem to be addressed (target case) in order to propose a suitable solution to improve the safety level of the software concerned. More specifically, these two tools are complementary and aim to suggest risks that are not taken into account during safety analyzes (automatisms level and software level) and therefore contribute to the search for the most appropriate preventive measures for to guard against a particular risk. The approach adopted to design and implement the tools ACASYA and SAUTREL is articulated around two main activities. The first activity is to extract, formalize and archive potential accident scenarios to develop a standard case library covering the entire safety problem. These dangerous situations are archived in a database called “Historical Scenarios Knowledge Base” (HSKB). The second activity aims at exploiting this stored historical knowledge (HSKB) in order to develop a safety analysis know-how that can help the experts to judge the comprehensiveness of the safety analyzes. This second activity is essentially based on the use of machine learning techniques.

6. ACASYA: a tool for analyzing and evaluating functional safety analysis

The choice of a learning system adapted to an industrial application is generally based on the identification of the needs, the characteristics of the available knowledge as well as on the definition of the expected performances of the learning system. For the safety problem and the certification of rail transport, the knowledge acquisition phase identified some 80 accident scenarios relating to the risk of collision. This set of scenarios is grouped by the safety expert into nine classes of scenarios such as the redundancy switching class and the initialization class. These scenario classes are archived in the Historical Knowledge Base of the Scenarios (HKBS). This base of learning examples is not completely representative of the field of railway safety and is tainted with "noisy" data. The objective of the study is to operate by machine learning on this basis in order to reproduce the activities of classification, evaluation and generation of potential accident scenarios involved in the evolutionary, intuitive and creative approach of the expert. In fact, in the presence of a new example of a scenario proposed by the manufacturer, the certification expert endeavors to classify it in an existing accident family while ensuring that this potential scenario takes into account all the breakdowns or possible failures. To identify the activity of finding failures likely to cause a situation of insecurity (or hazardous situation contrary to the safety), the mechanism of learning must produce a base of rules of the form: “if symptoms then failures”, exploitable by an inference engine of an expert system. After briefly recalling the essential characteristics of the field and introducing our approach for the development of a tool to assist in the analysis and evaluation of functional safety, we now justify the choice of systems and learning algorithms selected with reference to all the

properties required by the tool to help the analysis of rail safety and to the current offer perceived through the literature review. The properties imposed on the rail safety analysis tool are presented below:

- Similarity-based learning (SBL): remember that the SBL method is characterized by the availability of a large number of examples supplying the learning system and the lack of knowledge on the field (or weak knowledge). Given the acquired safety knowledge that is essentially accident scenarios, the choice of the SBL is justified.
- Symbolic-digital processing of data: the data processing chosen is of a symbolic-digital nature. It combines the efficiency of digital processing that allows operating in the presence of noisy and incomplete safety data and the explicability of the symbolic processing necessary for the user to understand the knowledge produced.
- Classification learning and empirical regularity learning: two learning strategies are required to ensure the two activities involved in the certification process: classification of accident scenarios and detection of empirical regularities to build knowledge bases exploitable by an expert system.
- Incremental production of conjunctive descriptions of object classes and rule generation: Classification activity requires non-monotonous incremental learning of conjunctive descriptions of accident scenario classes. The scenario evaluation activity requires the production of production rules to assist in and failures dangers recognition.
- Rules structuring: the learning system must generate, not isolated rules with a single inference step, but a system of structured rules that allows the formation of a deductive reasoning essentially taking into account the orientation of the rules: of the symptoms to the causes (failures).
- Non-monotonous learning: incrementality is an indispensable property for dealing with the evolving knowledge characteristic of the field of railway safety. It must be non-monotonous to ensure the possible questioning of knowledge previously learned. To guarantee the non-monotony of knowledge it is necessary to integrate means allowing to stabilize them and consequently to ensure the convergence of the system.
- Expert / System Interactivity: Inductive learning is inherently uncertain and produces plausible knowledge that the domain expert must validate. The intervention of this latter should not be limited, as in most learning systems, to the provision of learning examples, but should also focus on the control of learned knowledge. The system, meanwhile, must argue its reasoning and decisions. This "interactive" or "supervised" learning promotes the acquisition of new knowledge. The association of the domain expert at each stage of the learning process requires the development of a user-friendly human-machine interface.

All of these properties are indispensable for the new and complex industrial application of rail transport certification. It can be seen that none of the studied learning systems alone satisfies all these properties. However, if we break down our problem by distinguishing the classification activity from the evaluation activity of the accident scenarios, we can consider using the “Charade” [46] system for generation of production rules, but we are forced to develop a new classification system for accident scenarios.

6.1. Rationale for choosing the rule learning system: “Charade”

“Charade” [46] not only allows to generate a rule system structured and exploitable by an

inference engine of an expert system, but also to complete the description of the examples provided by the expert to take into account possibly noisy data such as the examples accident scenarios. It makes it possible to simultaneously learn certain logical rules and uncertain rules modulated by a likelihood coefficient. Finally, its major originality lies in its flexibility and its translation of the KBS functionalities that one wants to obtain thanks to the constraints that it implements. All of these benefits come at the cost of learning that cannot be incremental. However, at the level of the development of the evaluation rules of the accident scenarios, the structuring of the rules has priority over the incrementality.

6.2. Need to develop a new classification learning system: "Clasca"

The analysis of the existing works with regard to the properties expected for the classification activity reveals shortcomings. The learning system that comes closest to the classification solution is the ID3 [47] algorithm and its derivatives. Nevertheless, these learning systems require that the examples to be classified are all available from the start of the learning phase. In practice, and particularly in the field of railway safety, it is difficult to obtain an exhaustive list of examples unless considerable time is spent in the data acquisition phase with the experts. This is all the more true as one is in the presence of an evolutionary domain. In addition, the internal learning mechanisms of the majority of classification systems are not accessible to the domain expert. Designing a learning mechanism for which a prominent place is left to the expert to judge, semantically, the quality of the knowledge produced is an interesting advance. Indeed, an apprenticeship supervised by the expert is in itself an approach likely to bring out knowledge that, initially, was not necessarily obvious or even consciously present in the expert's mind. In view of these remarks, we propose to start the learning phase with a lot of examples pre-classified by the expert and not representative of the field, without obliging the expert to list all the examples but by involving it throughout the learning process to improve the knowledge acquired. As a result, the semantics of knowledge are taken into account. Then, the system is evolved with each new example of scenario provided by the expert to incrementally form conjunctive descriptions of potential scenario classes, comprehensible by the expert and compatible with the "Charade" system. This approach, which lies between non-incremental learning systems requiring the presence of all the examples to be classified and those incrementally dealing with the examples one by one, is the subject of the "Clasca" system, conceived and detailed later in this article. In the preceding paragraphs, we have presented the field of safety and certification of rail transport, the limits of the usual means of acquiring knowledge as well as the need to use machine learning to better understand the process of transferring certification expertise. The rest of this article proposes the different stages of design and realization of the "Acasya" system of assistance in the analysis and evaluation of functional safety. This is essentially based on the joint use of the "Charade" and "Clasca" modules previously identified.

7. Detailed description of the safety assessment methodology: "Acasya tool"

The rail safety analysis and assessment methodology is organized in ten steps. The first seven steps are carried out by the scenario classification module (Clasca) and the last three steps concern the scenario evaluation module which is based on the "Charade" rule learning system:

1. Acquisition of safety knowledge

2. Pre-design: Parameters and learning constraints
3. Learning: Induction of description of classes of scenarios
4. Classification of a new example of a scenario
5. Validation of knowledge learned by the system
6. Study of convergence of the learning system
7. Update of the HSKB database
8. Learning the Summarized Failures (SF) recognition functions: produced by “Charade”
9. Deduction of SFs who are to be considered in the manufacturer's scenario
10. Validation by the safety Expert

Very schematically, the first module of classification is concerned with carrying out a learning operation of the concepts and more precisely with descriptions characteristic of the historical scenarios resulting not only from the experience of the safety experts, but also from the safety files of the safety systems rail transport already certified and commissioned in France. The second evaluation module looks for regularities found in the scenario database and more precisely in each scenario class (identified by the previous module) in order to create a rule base for recognizing the presence of potential dangers involving the overall safety of the system. The first level of classification was presented in detail in [48,49] and the second level of evaluation is presented in [50–52]. Therefore, we will present in the rest of this paper only the outline and the methodological indications allowing the reader to make the most of the articulation between these two learning modules. This new approach is presented below in three main phases: Acquisition of knowledge, classification and evaluation.

7.1. Acquisition and modeling of safety knowledge

The knowledge acquisition phase has resulted in the development of a Historical Scenarios Knowledge Base (HSKB) that includes eighty scenarios of accidents or incidents related to collision risk such as the problem of redundancy switching, Penetration on a busy canton, Improper Initialization, Mating failure of elements, Inversion of order of elements, Failure to record after a needle, Crossing a breakpoint in manual driving. All eighty scenarios were subsequently grouped by safety experts into several classes or family of scenarios such as the class “Redundancy switching”, the class “Initialization sequence”, the class “Location of trains” or the class “Emergency braking management”. This HSKB, which forms the basis of the learning examples, will be exploited by the CLASCA learning algorithm in order to find the membership class of a new scenario proposed by the transport system manufacturer. This HSKB database will also be exploited by the CHARADE learning algorithm to produce rules necessary for learning the Summary Fault (SF) Recognition functions.

An accident scenario describes a combination of circumstances which can lead to an undesirable, perhaps even hazardous, situation. It is characterized by a context and a set of events and parameters. Examination of the concept of scenario revealed two fundamental aspects. The first is "static" and characterizes the context. The second is "dynamic" (modeled by a Petri net) and shows the possibilities of change within this context, while stressing the process which leads to an unsafe situation. The “static description” of a scenario is used by the first automatic learning module namely CLASCA which is dedicated to the classification of accident scenarios. The formalism used for the static description of a potential accident scenario is that of a “descriptive form” in which several

essential descriptive parameters are described in terms of attribute / value pairs. The attributes correspond to the eight characteristic parameters of a scenario (Type of block, Hazards (Risks), Hazard related functions, Incidental Functions, Elements Involved, Geographical zones, Summarized Failures, Adopted solutions). Each attribute is associated with a list of possible values (Table 1). This "descriptive form" was subsequently used as the basic form for the acquisition of the eighty scenarios. In summary, the static description of a scenario led to the definition of a first description language for the example scenarios. This is a classical representation by Attribute-Value couples. The scenarios which have been collected together so far in the historical knowledge base relate to the collision problem and have been constructed on the basis of the safety dossiers of rail transport systems French: VAL, POMA 2000, MAGGALY and TVM430 (Nord TGV) systems and the know-how of experts. More precisely, the level of detail which is required in system description in order to formalize the scenarios relates essentially to the general specifications of the system, the functional specifications and functional safety analysis.

Table 1. Extract from the formalism elaborated for the representation of accident scenarios.

	Attributes	**Possible values**
Symptoms	Type of block (TB)	Fixed blocks
		Moving blocks
	Hazards (Risks) (H)	Collision
		Derailment
		Etc.
	Hazard related functions (HRF) *These are protective functions which are intended to remove the hazard or make it acceptable to the user.*	Management of automatic driving
		Localization des trains
		Initialization
		Etc.
	Incidental Functions (IF) *These are functions which are related to the operation of the system and which can promote the occurrence of a scenario.*	Route management
		Traffic control
		Communication (transmission)
		Etc.
	Elements Involved (EI)	Instructions (consistency, vigilance)
		Operator at the control centre (CC)
		Etc.
	Geographical zones (GZ)	Terminus
		Station
		Etc.
Causes	**Summarized Failures(SF)** *SF is a* generic failure produced by the combination of a set of basic failures which has the same effect on the performance of the system.	Element and target in opposite direction
		Train reversing into an occupied block
		Collision avoidance transmitter failure
		Etc.
Remedies	Adopted solutions (AS)	Prohibit change of route if the approach area is occupied
		Increase the length of the Canton
		Etc.

7.2. Classification of accident scenarios

The first level of analysis relates to finding the class to which a new scenario which has been suggested by the manufacturer belongs. The purpose behind this is to provide the expert with historical scenarios which are partially or completely similar to the new scenario. This mode of reasoning is analogous to that which experts use when they attempt to find similarities between the situations which have been described by the manufacturer's scenarios and certain experienced or envisaged situations involving equipment which has already been certified and approved. CLASCA is a learning system by researching classification procedures. It is inductive, incremental and dedicated to the classification of accident scenarios. Learning in CLASCA is on the one hand non-monotonous to take into account the noisy and incomplete data relating to the scenarios and on the other hand supervised to allow the expert to correct and complete the initial knowledge and / or produced by the system. CLASCA incrementally develops conjunctive descriptions of historical scenario classes in order to characterize a set of insecurity situations and to identify a new scenario submitted for evaluation to the experts. The classification of a new scenario includes the following two major phases [52]:

– A characterization (or generalization) stage for constructing a description for each class of scenarios. This stage operates by detecting similarities within a set of historical scenarios in the HSKB which have been pre-classified by the expert in the domain. Each description which is learnt is characterized by a combination of three elements: (<Attribute> <Value> <Frequency>). The frequency of appearance is computed for each descriptor <attribute/value> (Formula 1). The objective is to determine the frequency of occurrence $\boldsymbol{\tau}$ in a selected example class C_k of the attribute A of rank n. $\tau_m^n(C_k)$ denotes the probability that the attribute A_n takes the value V_m^n in the example e_p^K and corresponds to the occurrence frequency of the value in class C_k.
– A deduction (or classification) stage to find the class to which a new scenario belongs by evaluating a similarity criterion. The descriptors of the new scenario are compared with the descriptions of the classes which were generated previously. In this stage a new example of a scenario is assigned to an existing class C_k. The classification phase of a new example of accident scenario requires the definition of a classification parameter called “adequacy rate” (T_{ad}) which measures the degree of resemblance between the new example E_i and each of the classes C_k of pre-existing scenarios. This T_{ad} is characterized as follows: (Formula 2). It adequacy rate (T_{ad}) based on statistical calculations is purely digital. We propose to refine it to take account of the semantics of the domain of application. The idea consists in extracting from the set of descriptors identified with the experts, the list of descriptors relevant to characterize each class of examples. The descriptors acquired and specific to each class are called “key descriptors”. For example, for the class “initialization sequence”, three key descriptors were defined by the expert: location of the trains, initialization and safety instructions. This point of view makes it possible to define a second rate of adequation which reflects the semantics of knowledge: (Formula 3). The combination of these two adequacy rates (2) and (3) ultimately leads to the definition of a rate to measure the adequacy between a new example E_i and a class C_k, taking into account both the statistical aspects and semantics of the data (Formula 4). λ is a smoothing coefficient that can be adjusted experimentally or proposed by the domain expert to take account of his deep convictions. It makes it possible to give more or less importance to statistical or semantic processing. For example, if $\lambda = 0$ the matching rate is purely semantic and if $\lambda = 1$, it is purely statistical. The two types of treatment are taken into account equally in the case where $\lambda = 0.5$.

$$\tau_m^n(C_k) = \frac{\sum_{p=1}^{\text{Card } C_k} D_m^n(e_p^k)}{\text{Card } C_k} \quad (1)$$

$$T_{ad\,1}(E_i, C_k) = \frac{\sum_{(m,n)\,/\,\tau_m^n(C_k)\ ?sd} D_m^n(E_i) \times \tau_m^n(C_k)}{\sum_{(m,n)\,/\,\tau_m^n(C_k)\ ?sd} \tau_m^n(C_k)} \quad (2)$$

$$T_{ad\,2}(E_i, C_k) = \frac{\sum_{(m,n)\,/\,(A_n, V_m^n)\ \text{soit "cl? de}\ C_k} D_m^n(E_i) \times \tau_m^n(C_k)}{\sum_{(m,n)\,/\,(A_n, V_m^n)\ \text{soit "cl? de}\ C_k} \tau_m^n(C_k)} \quad (3)$$

$$T_{ad}(E_i, C_k) = \lambda\ T_{ad\,1}(E_i, C_k) + (1-\lambda) T_{ad\,2}(E_i, C_k) \quad (4)$$

$$SS(C_K, n) = (1 - \alpha\, e^{\beta\,(1 - \text{Card } C_K)})\,(1 - \gamma\, e^{\delta\,(n_o - n)}) \quad (5)$$

The integration of a new example in a class causes the refreshing of the frequency of appearance of the descriptors. In this context, the unavoidable presence of "noise" makes non-monotonic learning necessary so that the frequency of appearance of a descriptor can increase or decrease depending on the influence of the new scenarios on the consistency of the class. To solve this problem of convergence, we agreed to change the value of the similarity threshold "SS" throughout the classification cycle, so as to be more and more "demanding" as the growth of the Cardinal of the class considered. This point of view has led to the definition of two types of convergence: the “internal convergence” that aims at the stability of knowledge within a class and the “global convergence” that ensures the stability of knowledge for all classes. These two types of convergence are encompassed in a broader definition called “enhanced internal convergence” (formula 5). SS (C_k, n) similarity threshold, increases monotonically as a function of Card C_k and n and tends to 1. It is updated with each addition of an example in a class. It should be noted that the values of α, β, γ and δ can be set differently from one class to another. β and δ act on the learning time and consequently on the speed of convergence. These are two mitigating factors of convergence. The modularity of β and δ allows the user to evolve at will his system and ensure its convergence. A scenario classified by the system, judged relevant and validated by the expert will subsequently be integrated into the HSKB database. This is a phase of updating the data and therefore learning new scenarios of potential accidents. This initial level of processing not only provides assistance to the expert by suggesting scenarios which are similar to the scenario which is to be dealt with but also reduces the space required for evaluating and generating new scenarios by focusing on a single class of scenarios C_k.

7.3. Evaluation of accident scenarios

The second level of treatment considers the C_k class previously identified by the classification module in order to evaluate the consistency and completeness of the new accident scenario (scenario of the manufacturer that the expert seeks to examine). The method of analysis and evaluation of safety is centered on the summarized failures (SFs) which are involved in accident scenarios capitalized. An accident scenario describes a set of circum- stances that can lead to a dangerous situation. It is characterized by a context and a set of parameters, in particular SF, risk (hazard), actors involved, incidental functions, and geographic zone. An SF is a generic failure produced by the combination of a set of basic failures which has the same effect on the performance of the system. Each scenario brings into play one or more SFs. A list has been compiled of the SFs involved in all the scenarios which have been collected so far.

The purpose is to automatically generate a recognition function for each SF associated with a scenario class. The SF recognition function is a production rule which establishes a link between a set of facts (parameters which describe a scenario or descriptors) and the SF fact. What is involved here is logical dependence, which can be expressed in the following form: IF Type of block (TB), And Hazard (H), And Hazard related functions (HRF), And Geographical zones (GZ), And Elements involved (EI), and Incident functions (IF) then Summarized Failures (SF).

This phase of learning attempts, using the base of 80 examples which was formed previously, to generate a system of rules. The conclusion of each rule which is generated should contain the SF descriptor. In this context, it has proved to be inevitable to use a learning method which allows production rules to be generated from a set of historical examples (or scenarios). The specification of the properties required by the learning system and a review of the literature has led us to choose the CHARADE mechanism [46]. CHARADE ability to generate automatically a system of rules, rather than isolated rules, and its ability to produce rules in order to develop SF recognition functions make it of undeniable interest. CHARADE is a learning system whose purpose is to construct knowledge based systems on the basis of examples. It makes it possible to generate a system of rules with specific properties. Rule generation within charade is based on looking for and discovering empirical regularities which are present in the entire learning sample. Regularity is a correlation which is observed between descriptors in the base of learning examples. If all the examples in the learning base which possess the descriptor d1 also possess the descriptor d2 it can be inferred that d1→d2 in the entire learning set (Figure 1).

Thereby, a base of evaluation rules can be generated for each class of scenarios. The evaluation of a scenario involves two modules [52]:

- A mechanism for learning rules CHARADE which makes it possible to deduce SF recognition functions and thus generate a base of evaluation rules,
- An inference engine which exploits the above base of rules in order to deduce which SFs are to be considered in the manufacturer's scenario. The SF deduction stage requires a preliminary phase during which the rules which have been generated are transferred to an expert system in order to construct a scenario evaluation knowledge base.

The aim of the evaluation module is to compare the list of SFs which are suggested in a manufacturer scenario to the list of stored historical SF (in the rule base of the expert system) in order to stimulate the formulation of hazardous situations which have not been anticipated by the

manufacturer. This evaluation task draws the attention of the expert to any failures which have not been considered by the manufacturer and which might jeopardize the safety of the transport system. It may thus promote the generation of new accident scenarios.

If Elements involved = mobile operator,
Incident functions = instructions
Elements-involved = operator in CC.
Then Summarized failures = SF11 (Invisible element on the zone of completely automatic driving),
Elements involved = AD with redundancy,
Hazard related functions =train localization,
Geographical zones = terminus.
[0]

Figure 1. A sample of some rules generated by CHARADE.

8. SAUTREL for the help to the evaluation of the safety of the critical software

Recall that our research focuses on the evaluation of two complementary safety analyzes: Functional Safety Analysis (FSA) and Software Errors and Effects Analysis (SEEA). We have just presented the tool “Acasya” for the help to the FSA. This paragraph is devoted to the tool “Sautrel” for the help to the evaluation of the safety of the critical software and in particular for the evaluation of the SEEA. Generally preliminary hazard analysis (PHD) can identify all potential accidents of the system such as collision, derailment of a train. As for the FSA, for each potential accident, it proposes to provide the functions and equipment needed to guard against these accidents. Our approach to safety assessment is organized around two closely related phases. To improve the FSA we proposed a new approach based on learning concepts and learning rules to generate potential hazards (SF) not considered by the system designer. This approach was the subject of the "Acasya" tool. In the face of a particular danger, constituting a new situation of insecurity, the "Sautrel" tool seeks to identify the solutions (or recommendations) necessary to cover the danger previously generated. This second evaluation phase, which is based on the use of case-based reasoning (CBR), aims to look for the most similar and analogous case in the historical accident and incident database (source case base) the problem to be addressed (target case) in order to propose the appropriate measures to avoid the occurrence of such a danger and consequently of a potential accident. Before detailing the main functionality of the tool “Sautrel” it should present the CBR.

8.1. Case-based reasoning (CBR)

The CBR is generally interpreted as an important process for solving new problems based on finding similar solutions to the problems of the past. It is part of a behavior commonly used in solving everyday human problems. Indeed, all human reasoning is generally based on past cases

lived personally. The CBR considers reasoning as a process of remembering a small set of practical situations. The cases, it bases its decisions on the comparison of the new situation (target cases) with the old (reference cases). The general principle of CBR is to treat a new problem (target case) by remembering similar past experiences (source cases). This type of reasoning rests on the assumption that if a past experience and new circumstances are sufficiently similar, then everything can be explained or applied to past experience (source cases) and remains valid when applied to the new situation which represents the new problem to solve. For example, in the field of technical or medical diagnostics, the expert in the field, faced with the symptoms observed, he often proceeds by analogical reasoning by referring to past historical cases to quickly explore and search for the causes of a risk of accident or illness (for the doctor) in order to propose a remedy for this new undesirable situation. CBR is an approach to problem solving that emphasizes the role of prior experience during future problem solving (i.e., new problems are solved by reusing and if necessary adapting the solutions to similar problems that were solved in the past). Very schematically, in the context of the CBR, a case is considered a problem with his solution as well as procedures allowing a justification of the decisions made on the way the solution was generated. The work of Aamodt [53], Harmon [54], Kolodner [55], Leake [56], Mott [57], Pinson [58] and Slade [59] provide a fairly complete retrospective of the evolution of case-based reasoning research (CBR).

8.2. Proposal for a method of assessment of critical software safety based on the CBR

In order to show the interest of machine learning and more precisely CBR in the field of the safety of railway transport, we have developed a tool called “Sautrel”. This tool helps safety experts in their SEEA document analysis and assessment tasks. The design and implementation of this tool required the following three major phases [60]:

– Acquisition and modeling of knowledge related to SEEA. This analysis and abstraction stage resulted in the production of formalism for SEEA which takes account of the practices and our experience in the field of railway safety. This model is based on eight characteristic parameters: The investigated system, the investigated subsystem, the investigated module, the envisaged error (family, class, type), the safety criterion infringed by the error, the feared hazard, the type and severity of possible damage and finally the means of detecting the error and protecting against it.
– Using the above model we built up a library of 250 cases (examples). These historical examples of SEEAs were drawn from two guided transport systems: MAGGALY and the TVM 430 for the Nord TGV.
– Development of the “Sautrel” tool. The mock-up has four main modules: A man/machine interface for inputting, updating and consulting knowledge relating to SEEA, a representation and acquisition module for SEEA sheets, a knowledge base containing 250 examples of SEEA (experience base), and a case-based reasoning process (implemented by the Recall software). The main components of this CBR process are a mechanism which indexes (or characterizes) target cases and a mechanism which finds similar cases (reference cases) and collects them together.

The "Sautrel" tool requires the following steps [60,61]

1. Acquisition and modeling of knowledge,
2. Definition of the description language of the SEEA examples,
3. Development the SEEA case base,
4. Parameterization and Calibrating of the CBR process,
5. Entering the new SEEA target case for evaluation,
6. Indexing of the SEEA case base,
7. Extraction of similar SEEA cases,
8. Adaptation of extracted cases (source cases),
9. Updating the SEEA base.

8.3. Acquisition and modeling of knowledge

This paragraph presents the results of the phase of formalization and acquisition of the knowledge necessary for the development of a historical case base (experience feedback) in order to capitalize and perpetuate the knowledge related to the SEEA. The first step of the study is devoted to the research and identification of descriptors and characteristic parameters to represent and formalize the SEEA. After a second step of data collection necessary to list the possible values taken by each parameter (or descriptor), the third step proposes, a formalism of representation of documents SEEA. Finally, on the base of this formalism, which constitutes the basic language of SEEA representation, the fourth stage of the study focuses on building the case base that currently comprises 224 cases, each of which represents a particular situation that is contrary to safety (Problem) and one or more preventive measures or corrective measures to guard against, avoid, reduce, or permanently eliminate the potential risk envisaged (Solution).

To leverage knowledge of SEEA (or historical cases), it is necessary to adopt a model (or formalism) that is generic enough to cover as much as possible SEEA documents (or files) from several more or less different transport systems. To build this model and in order to show the feasibility of the study, we examined the SEEA relating only to two rail transport systems already certified and put into circulation in France: the automated system MAGGALY and the system TVM (track-to-train transmission) of the LGV Nord. It is important to emphasize that each SEEA file is specific to a particular system and therefore it is necessary to perform sufficient analysis and abstraction work to cover the majority of systems. Indeed, this analysis presents some difficulties, since from one manufacturer to another, or even from one system to another, the formalism, the terminology or the level of deepening of the analysis implemented are different. At the end of this review, we finally proposed a first SEEA representation model that relies heavily on the manufacturers' practices and our experience in the field of railway safety. This formalism is based on eight characteristic parameters: Studied system, subsystem studied, module studied, error envisaged (family, class, type), safety criterion not respected by the error, dreaded event, type and gravity of the damage, barrier and means for detecting the error. This model proposes a methodological framework for preparing SEEA files and thus contributes to ensuring the quality of future analyzes. An excerpt from this formalism is presented in Figure 2. On the basis of this representation model of the SEEA forms, we have created a library of 224 typical cases.

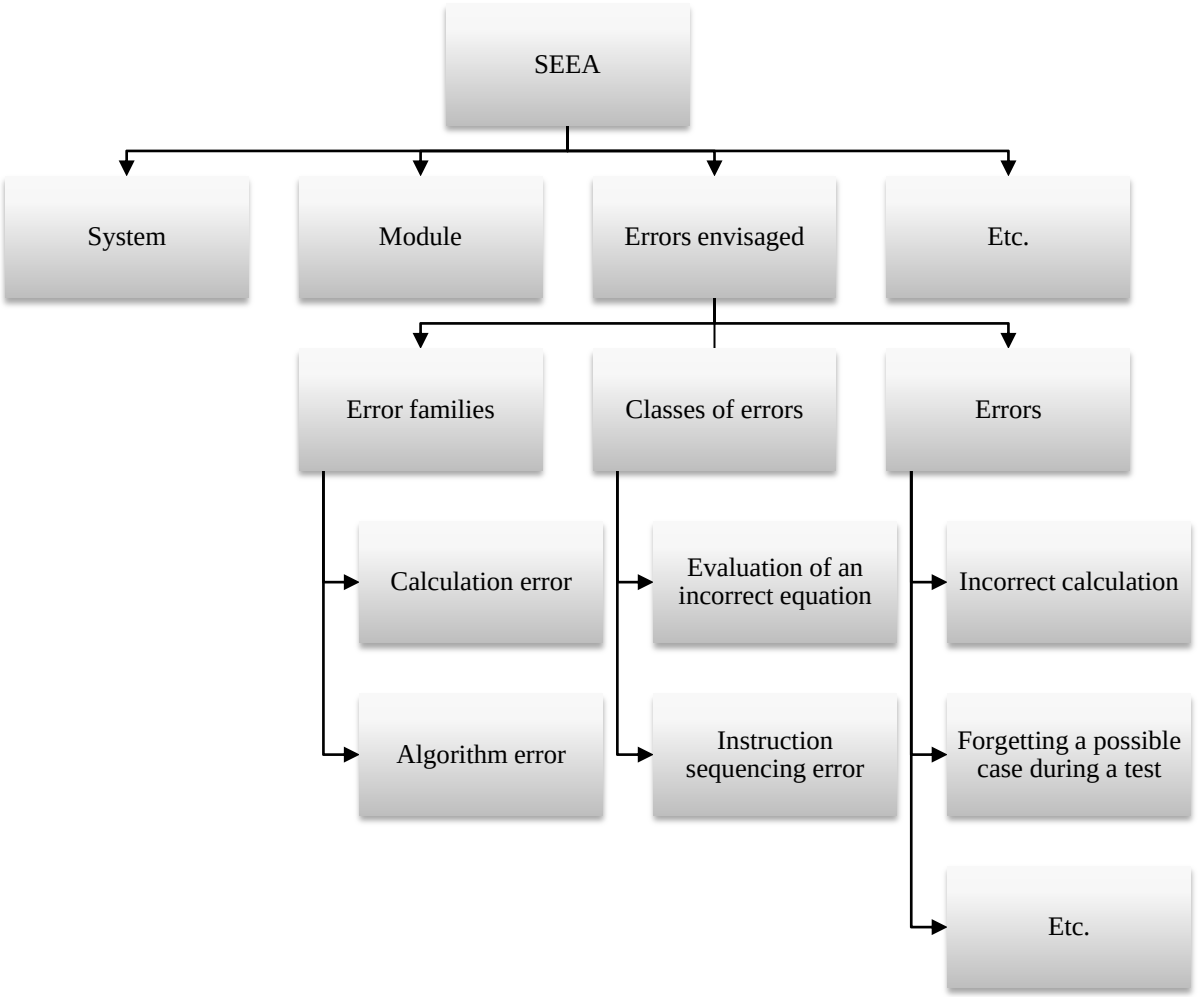

Figure 2. Extract from the formalism elaborated for the representation of records SEEA.

8.4. Definition of the description language of the SEEA examples

This step allows you to enter the description language of an SEEA based on the eight descriptors listed above. A descriptor is a couple (attribute, value). All attributes are symbolic. Three types of descriptors could be distinguished: Enumerated descriptors, multi-valued descriptors and unknown descriptors.

8.5. Developing the SEEA case base

It's about creating cases by assigning a value to each attribute of the description language. This case base may subsequently be modified or consulted. The acquisition of the target case is done by entering the value or values of the different attributes. During this case base construction step, the concept descriptor "dreaded event" is left unknown because it represents the solution we are looking for in the case base.

8.6. Parameterization of the CBR process

During this step, the user must set different parameters to configure the CBR process. These choices concern both the descriptor that will represent the solution of the problem and the strategies of indexing, matching or adaptation. During this step, the user must set the following parameters:

- The descriptor "concept": The user must choose from all the descriptors the one that will represent the solution of the problem. In our example, the descriptor "concept" is the descriptor "dreaded event". The problem, meanwhile, will be characterized by all the other descriptors.
- Indexing strategies: The tool offers several strategies for prioritizing memory. The user can set this hierarchy by sorting the descriptors or trimming the hierarchy. In our example, we construct the hierarchy by taking into account all the descriptors and by imposing the descriptors "studied system" and "studied subsystem", in this order, as first and second level of the decision tree. Then, the choice between the remaining descriptors for the next levels will be done by a decision tree classification algorithm: Quinlan ID3 algorithm [47].
- Matching strategies: The user can intervene in several ways in calculating the similarity between two attributes. It can possibly specify the descriptors which will not have to be taken into account during the computation. It can also give a weight vector to indicate the relative importance of a descriptor over others. In our example, we chose to extract only the 10 most similar cases, and to give a weight equivalent to all the descriptors.
- Adaptation strategies: To date, the tool does not offer a real adaptation method, but allows the user to program his own methods by demons. Currently, this adaptation can be done either implicitly by the safety domain expert, by comparing cases similar to the target case, or by the voting technique. In this second case, the value of the attribute to be adapted is calculated on all the similar cases by a vote weighted by the percentage of similarity of each case. For example, if a case C has 3 descriptors of which 2 are 100% similar to the target case and the third descriptor has no similarity (0%), then case C will be similar with the target case at 66%. If all the descriptors are of equal weight: (100 × Descriptor weight 1 + 100 × Descriptor weight 2 + 0 × Descriptor weight 3)/3 = 66.

8.7. Entering the new SEEA target case for evaluation

The acquisition of the target case is done by entering the value or values of the different attributes but leave the concept descriptor "dreaded event" unknown because it represents the solution we are looking for in the source case base.

8.8. Indexing of the SEEA case base

After developing the SEEA case representation mode, i.e. the description of the problem and the solution in the form of descriptors (attribute/value), it is then necessary to build a model for organizing and indexing the memory. This model is essential in the search for similar cases and must have certain qualities. Knowing that the research phase of similar cases must keep a constant complexity as the case base is filled; it is wise to consider a solution to quickly find similar cases. To apprehend this problem, we use the indexing method where each node of the tree corresponds to a question on one of the indexes and the threads of the tree correspond to the different answers. An index represents the elements discriminating the cases and has two fields: its name and its value. To

ensure a minimum of efficiency, the tree, which is dynamically built, must ask the questions in the right order and be as shallow as possible. The best way to build it is to use the decision tree method. Decision tree consists of nodes corresponding to the attributes of the selected objects and branches characterizing the alternative values of these attributes. The leaves of the tree represent the sets of objects of the same class of objects. The construction of decision trees is a top down generalization approach. The ID3 of QUINLAN algorithm [47] is a typical case of a downward approach. ID3 uses a heuristic search strategy, according to the gradient method, by optimizing a numerical criterion called gain of information which is based on the entropy of SHANNON developed in the early 1940s by Claude Shannon [62].

From:

- A set of exclusive classes {C1, C2, ... C_k};
- A set of examples {E1, E2, ... E_n} represented in the form of pairs (attribute/value) and partitioned in classes C_i;

ID3 produces a decision tree that allows to recognize (or classify) all the examples E_i.

This tree can then be used to generate classification rules.

QUINLAN's method consists in successively testing each attribute to know which one to use first in order to optimize the gain of information. That is, the attribute that best distinguishes between examples of different classes. This principle has been applied in many cases and has contributed to the development of several expert systems, essentially dedicated to diagnosis. Subsequently, work was devoted to improving the principle of construction of the decision tree and in particular reducing the size of the tree, improving the selection strategy (which is based in ID3 only on the attribute) by proposing a selection based on both the pair (attribute/value) or the improvement of the representation mode of the examples, by using a representation based on diagrams (frames). Used in a variety of fields such as data mining, business intelligence, medicine, safety, etc., the decision tree is a decision support tool that represents a set of choices in the form of graphical data (tree). In our case of application to SEEA, we use the classification algorithm ID3. During this indexing or prioritization step, the user selects the case base to index, and then starts the construction of the hierarchy. In our example (Figure 3), the first two levels of the hierarchy are constructed from the descriptors "studied system" and "studied subsystem". Here, the third level deals with the descriptor "Severity of the damage".

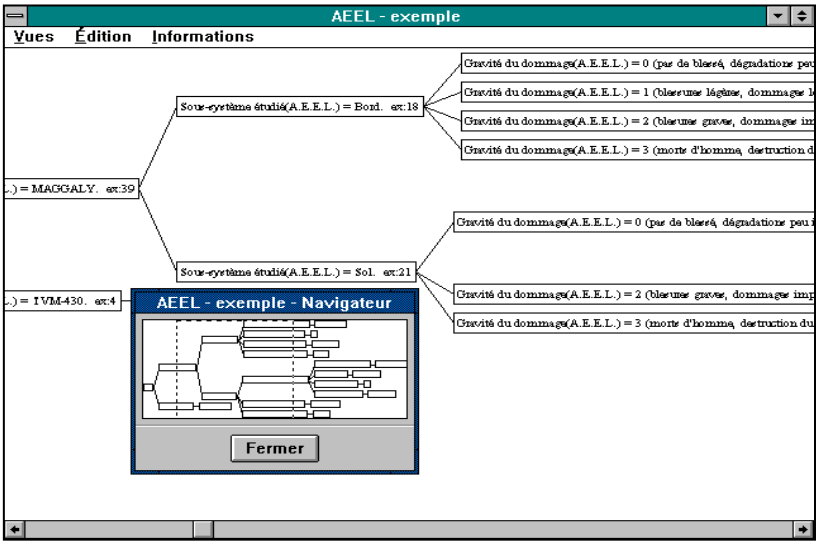

Figure 3. Example of the instances base hierarchy.

8.9. Extraction of similar SEEA cases

The Before searching for similar cases, if some information is missing (for example, a value of an attribute not specified), it is possible to complete the knowledge acquisition phase by querying the domain expert. There are some learning tools to try to determine and correct this data. In our case of application, during the phase of acquisition and collection of SEEA data, particular attention was paid to this problem of noisy or inconsistent data. The search for SEEA cases similar to the target case, is broken down into two filtering and selection stages that use static and dynamic indexes. There are different ways to determine the characteristics of indexes: All characteristics, some characteristics, the most discriminating characteristics, etc. In our application we adopted a similarity search based on the set of characteristics. To find similar SEEA cases from the case database archived in memory (source cases or reference cases), several techniques can be used, such as the "Nearest Neighbor" algorithm whose objective is to measure the similarity between the problem (target case) and potential source cases. The comparison method is based on the indexes. Thus, from the similarity on each index, the algorithm generates the global similarity sought. Let's remember that the search for nearest neighbors, or k nearest neighbors commonly used in machine learning, consists of starting from a set of other points to find the nearest K (similar) points. Generally, to optimize this method, we use heuristics and selection strategies to quickly find the most useful cases to solve the problem. The cases that share the most important characteristics, the easiest cases to adapt or the most used cases are examples of heuristics. In our application example, from the historical case base (source cases), it is a question of finding the SEEA cases most similar to the SEEA cases to be evaluated (target case) and who share the most important characteristics.

8.10. Adaptation of extracted cases (source cases)

Suppose we found a similar case, so we reuse directly the solution he proposes to solve the problem (case target). In practice, it is often rare that we find a case identical to the problem, so it is necessary to adapt pre-existing solutions. Adaptation therefore consists of building a new solution from the target case and similar cases found. It is then necessary not only to look for the difference between the cases found (source cases) and the problem, but also to find the useful information to be transferred to the new solution. Generally, one distinguishes two types of adaptation: Transformational adaptation and derivative adaptation. In the first approach, it is a question of directly reusing the solutions of the past cases. This type of transformational adaptation does not tell us how the solutions of similar cases were generated. It is the role of derived adaptation that allows, for each case stored in the database, to explain the reasoning process leading to the solutions. In this case, the derivative adaptation consists in applying the same reasoning to the new problem by choosing the paths taken by the old solutions selected and thus avoiding any unsuccessful paths. In our application case, the ReCall tool used to demonstrate the feasibility of the proposed approach does not yet propose relevant adaptation strategies. To date, the adaptation phase is still assigned to the user and in particular to the safety expert. With the screen presented in Figure 4, the user can consult the value taken by the concept attribute "dreaded event" in each similar case and choose himself the value to give to the "concept" attribute for the target case. The user can also use the voting technique. In our example, the tool proposes a single value for the attribute "dreaded event": Train collision. Thus, the domain expert can adapt the most similar case (proposed by the tool) by assigning the "Feared Event" concept the value "Collision" as a solution to the problem.

Since the ReCall tool does not propose adaptation strategies, the adaptation phase is limited in our example to indicate the class of potential solution. The solution sought is therefore focused simply on the value of the concept “feared event” proposed by the tool: “collision”. Nevertheless, this knowledge is necessary to stimulate and assist the expert in his task of safety assessment. Indeed, faced with a new problem (scenarios of accident/potential incident) described by a set of characteristic descriptors, it is interesting to know the possible feared event or events (collision, derailment, electrocution, fall).

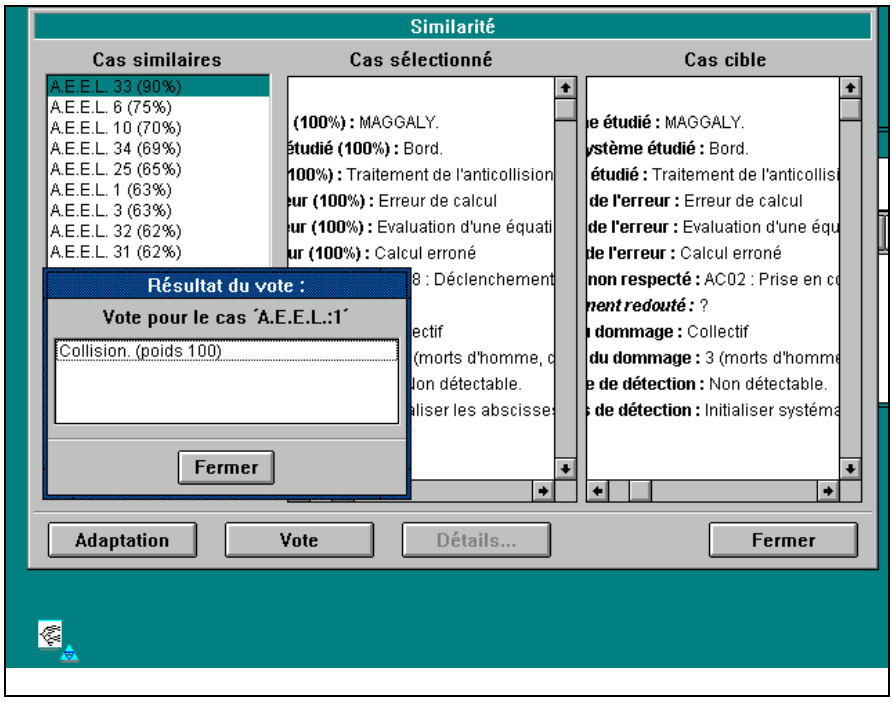

Figure 4. Example of the reference cases consultation and the vote technique use.

8.11. Updating the SEEA base

This last step of updating knowledge is to perform the automatic learning by adding the appropriate target case in the SEEA historical case base. In the ReCall software, this learning is not incremental since the new case will be integrated into the hierarchy without it being reconstructed. It is up to the user to take the initiative to relaunch the indexing of the case base. Therefore, during this phase of the CBR cycle, it is wiser that the new case with its new solution is validated by the domain expert before being added to the case base (source cases). In addition, it is interesting at the end of this learning phase to test the system by relying on the same problem that it has just treated to ensure that the system behaves as expected. Finally, it is essential to determine how to index this new case in the database without questioning the historical knowledge learned in previous phases and thus avoid new problems of inconsistency, redundancy, etc. In particular, the focus must be on this problem of incrementality. Should we adopt a monotonous incremental learning approach (accumulation of knowledge without questioning knowledge previously learned) or non-monotonous (examination of knowledge learned with each addition of new knowledge)? This is a problem that remains crucial in almost all machine learning systems. As part of our prototype of feasibility, this work has not yet completed.

9. Discussion

The knowledge acquisition phase ultimately culminated in the implementation of a conceptual FSA and SEEA representation model that provides a methodological framework for safety experts. Based on this model, we acquired 224 cases of SEEA and 80 accident scenarios involved in AFS (two historical databases for learning). These two learning bases are based on feedback from several rail transport systems put into service in France, such as the "Maggaly" system and the "VAL" system, which are fully automated, or the High Vessel Train (TGV-North).

When it comes to machine learning, our work is part of supervised learning. Indeed, the presence of the safety expert is essential to ensure effective and relevant learning. The domain expert is not only able to control, validate, adapt and complete the knowledge learned by the system, but also to adjust certain learning parameters. To demonstrate the feasibility of the proposed approach, we used three learning systems. The first "Clasca" which is an inductive and incremental learning algorithm allows grouping and classifying the historical accident scenarios. The second "Charade", which was kindly provided by the Prf. Jean-Gabriel Ganasca (LIP6-Jussieu-Paris 6), strives to look for regularities present in a class of accident scenarios (proposed by the "Clasca" module) in order to produce a rule base that can be exploited by a expert system. For the third case-based reasoning algorithm (CBR), we used a CBR generator called "ReCall" from ISOFT.

Despite the undeniable interest of these tools Clasca, Charade and ReCall, several shortcomings were noted during the evaluation phase. As stated above, for the Clasca system, it is necessary firstly to enrich the learning base so that it is representative of the field of railway risk management and secondly to implement a new Approach in order to apprehend the problem Inherent in the sensitivity of the system to the order of arrival of the example scenarios. With regard to the case-based reasoning system "ReCall", several shortcomings have been noted in particular for methods for calculating similarity, coping strategies and processing missing values (noisy data). Finally, the rule learning system "Charade", despite its undeniable interest, some rules generated are not of direct interest to assess safety. It is therefore essential to check, with the help of the domain expert, the veracity and the relevance of certain rules.

10. Conclusion

In this article, we characterized the safety domain and defined the objectives of our research. This study focused on the various safety analysis methods employed by the experts and in particular the functional safety analysis (FSA) and software error analysis (SEEA) methods. This research has also raised the problem of exploiting databases based on experience feedback on accidents and incidents. In spite of the undeniable interest of the usual methods, no method, alone, makes it possible to carry out an exhaustive safety analysis. For such methods, the completeness of the safety analysis remains largely based on human intelligence and intuition. Thus, after having recalled the essential characteristics of the field and introduced our approach for the development of a tool to help in the evaluation of safety, we justify the choice of the learning systems selected with reference to all the properties required by the safety tool and the current offer perceived through the bibliography. Referring to the bibliographic study, it can be seen that none of the learning systems considered alone satisfies all these properties. However, if we break down our problem by distinguishing the classification activity from the evaluation activity of the accident scenarios, we considered using the "Charade" system for the generation of rules, the tool "ReCall" for case-based reasoning (CBR), but one is forced to develop a new system called "Clasca" for the classification of

accident scenarios. These three learning systems are complementary for the development of a tool to assist in the analysis and evaluation of railway safety. The first classification module for accident scenarios is based on two main steps: 1) a characterization step (or generalization) for the construction of an intentional description of each of the scenario classes. This step operates by detecting similarities in a set of historical scenarios pre-classified by the domain expert. 2) a deduction step (or classification) for the search of the class of membership of a new scenario by evaluation of a criterion of adequacy. The second evaluation level, which is based on the joint use of the "Charade" system and an expert system, makes it possible to generate hazard recognition functions for each class produced by "Clasca". With the help of an expert system, these potential hazards are confronted with the hazards suggested by the manufacturer with a view to generating new dangers that could jeopardize the safety of the system. Faced with a particular danger constituting a new situation of insecurity, the CBR tool seeks to identify the solutions (or recommendations) necessary to guard against the danger previously generated. From the historical case base, it is a question of finding the most similar case which can solve the new problem to be treated (new danger) in order to propose the appropriate measures to avoid the occurrence of such a danger and thus, of a potential accident.

Currently, project is at the mock-up stage. Initial validation has demonstrated the interest of the suggested approaches, but improvements and extensions are required before they could be used in an industrial environment or adapted to other areas where the problem of investigating safety arises. These improvements include the improvement of the adaptation strategies of the solutions proposed by the system, the enrichment of the basis of accident scenarios to cover the whole problem and finally, it is necessary to construct an integrated version of a prototype in order to finalize the results of the demonstration model.

Conflict of interest

The author declares that there is no conflict of interest in this paper.

References

1 Hadj-Mabrouk H (2018) New approach of assessing human errors in railways. *Transactions of the VSB - Technical University of Ostrava, Safety Engineering Series* 13: 1–17.

2 Hadj-Mabrouk H (2019) Consideration of Human Factors in the Accident and Incident Investigation Process. Application to the Safety of Railway Transport. *J Ergon Adv Res* 1: 1–20.

3 Hadj-Mabrouk H (2016) Knowledge based system for the evaluation of safety and the prevention of railway accidents. *International journal of railway* 3: 37–44.

4 Bergmeir C, Sáinz G, Bertrand CM, et al. (2013) A Study on the Use of Machine Learning Methods for Incidence Prediction in High-Speed Train Tracks. *IEA/AIE 2013 Proceedings of the 26th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems* 7906: 674–683.

5 Fay A (2000) A fuzzy knowledge-based system for railway traffic control. *Eng Appl Artif Intel* 13: 719–729.

6 Santur Y, Karaköse M, Akin E (2017) A new rail inspection method based on deep learning using laser cameras. *International Artificial Intelligence and Data Processing Symposium (IDAP)* 16–17.

7 Faghih-Roohi S, Hajizadeh S, Núñez A, et al. (2016) Deep convolutional neural networks for detection of rail surface defects. *International Joint Conference on Neural Networks (IJCNN)* 24–29.

8 Ghofrania F, He Q, Goverde R, et al. (2018) Recent applications of big data analytics in railway transportation systems: A survey. *Transport Res C-Emer* 90: 226–246.

9 Thaduri A, Galar D, Kumar U (2015) Railway assets: A potential domain for big data analytics. *Procedia Comput Sci* 53: 457–467.

10 Attoh-Okine N (2014) Big data challenges in railway engineering. *IEEE International Conference on Big Data (Big Data)* 27–30.

11 Hughes P (2018) Making the railway safer with big data. Available from: http://www.railtechnologymagazine.com/Comment/making-the-railway-safer-with-big-data.

12 Hayward V (2018) Big data & the Digital Railway. Available from: https://on-trac.co.uk/big-data-digital-railway/.

13 Marr B (2017) How Siemens Is Using Big Data And IoT To Build The Internet Of Trains. Available from: https://www.forbes.com/sites/bernardmarr/2017/05/30/how-siemens-is-using-big-data-and-iot-to-build-the-internet-of-trains/#2b7a4b6e72b8.

14 Williams T, Betak J, Findley B (2016) Text Mining Analysis of Railroad Accident Investigation Reports. *Proceedings of the 2016 Joint Rail Conference*.

15 Brown DE (2016) Text Mining the Contributors to Rail Accidents. *IEEE Transactions on Intelligent Transportation Systems* 17: 346–355.

16 Li J, Wang J, Xu N, et al. (2018) Importance Degree Research of Safety Risk Management Processes of Urban Rail Transit Based on Text Mining Method. *Information-an International Interdisciplinary Journal* 9: 26

17 Williams T, Betakbc J (2018) A Comparison of LSA and LDA for the Analysis of Railroad Accident Text. *Procedia Computer Science* 130: 98–102.

18 Syeda K, Shirazi SN, Naqvi SA, et al. (2018) Big Data and Natural Language Processing for Analysing Railway Safety: Analysis of Railway Incident Reports. *Innovative Applications of Big Data in the Railway Industry* 240–267.

19 Van-Gulijk C, Hughes P, Figueres-Esteban M, et al. (2018) The case for IT transformation and big data for safety risk management on the GB railways. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 232: 151–163.

20 Syeda KN, Shirazi SN, Naqvi SAA, et al. (2017) Big Data and Natural Language Processing for Analysing Railway Safety. *Innovative Applications of Big Data in the Railway Industry. IGI Global Publishing* 240–267.

21 Ghomi H, Bagheri M, Fu L, et al (2016) Analyzing injury severity factors at highway railway grade crossing accidents involving vulnerable road users: A comparative study. *Traffic Injury Prevention* 17: 833–841.

22 Zhang X, Green E, Chen M, et al. (2019) Identifying secondary crashes using text mining techniques. *Journal of Transportation Safety & Security* 1–21.

23 Heidarysafa M, Kowsari K, Barnes LE, et al. (2018) Analysis of Railway Accidents' Narratives Using Deep Learning. *International Conference on Machine Learning and Applications (LCMLA)* 1446–1453.

24 Gibert X, Patel VM, Chellappa R (2017) Deep multitask learning for railway track inspection. *IEEE T Intell Transp* 18: 153–164.

25 Osman A, Hajij M, Bakhit PR, et al. (2019) Prediction of Near-Crashes from Observed Vehicle Kinematics Using Machine Learning. *Transportation Res Rec*.

26 Nakhaee MC, Hiemstra D, Stoelinga M, et al. (2019) The Recent Applications of Machine Learning in Rail Track Maintenance: A Survey. In: Collart-Dutilleul S., Lecomte T., Romanovsky A. (eds) *Reliability, Safety, and Security of Railway Systems. Modelling, Analysis, Verification, and Certification*. RSSRail 2019. Lecture Notes in Computer Science.

27 Zubair M, Khan MJ, Awais M (2012) Prediction and analysis of air incidents and accidents using case-based reasoning. *Third Global Congress on Intelligent Systems* 315–318.

28 Khattak A, Kanafani A (1996) Case-based reasoning: A planning tool for intelligent transportation systems. *Transport Res C-Emer* 4: 267–288.

29 Sadeka A, Smith B, Demetsky M (2001) A prototype case-based reasoning system for real-time freeway traffic routing. *Transport Res C-Emer* 9: 353–380.

30 Sadek A, Demetsky M, Smith B (1999) Case-Based Reasoning for Real-Time Traffic Flow Management. *Comput-Aided Civ Inf* 14:347–356.

31 Zhenlong L, Xiaohua Z (2008) A case-based reasoning approach to urban intersection control. *7th World Congress on Intelligent Control and Automation* 7113–7118.

32 Li K, Waters NM (2005) Transportation Networks, Case-Based Reasoning and Traffic Collision Analysis: A Methodology for the 21st Century. In: Reggiani A, Schintler LA (eds.), *Methods and Models in Transport and Telecommunications*, 63–92.

33 Kofod-Petersen A, Andersen OJ, Aamodt A (2014) Case-Based Reasoning for Improving Traffic Flow in Urban Intersections. *International Conference on Case-Based Reasoning* 8765: 215–229.

34 Louati A, Elkosantini S, Darmoul S, et al. (2016) A case-based reasoning system to control traffic at signalized intersections. *IFAC-Papers On Line* 49: 149–154.

35 Begum S, Ahmed MU, Funk P, et al. (2012) Mental state monitoring system for the professional drivers based on Heart Rate Variability analysis and Case-Based Reasoning. *Federated Conference on Computer Science and Information Systems (FedCSIS)* 35–42.

36 Zhong Q, Zhang G (2017) A Case-Based Approach for Modelling the Risk of Driver Fatigue. *International Conference on Intelligence Science* 510: 45–56.

37 Varma A, Roddy N (1999) ICARUS: Design and deployment of a case-based reasoning system for locomotive diagnostics. *Eng Appl Artif Intel* 12: 681–690.

38 Johnson C (2000) Using case-based reasoning to support the indexing and retrieval of incident reports. *Proceeding of European Safety and Reliability Conference (ESREL 2000): Foresight and Precaution, Balkema, Rotterdam, the Netherlands* 1387–1394.

39 Cui Y, Tang Z, Dai H (2005) Case-based reasoning and rule-based reasoning for railway incidents prevention. *Proceedings of ICSSSM '05. 2005 International Conference on Services Systems and Services Management* 2: 1057–1060.

40 Li X, Yu K (2010) The research of intelligent Decision Support system based on Case-based Reasoning in the Railway Rescue Command System. *International Conference on Intelligent Control and Information Processing* 59–63.

41 Lu Y, Li Q, Xiao W (2013) Case-based reasoning for automated safety risk analysis on subway operation: Case representation and retrieval. *Safety Sci* 57: 75–81.

42 de Souza VDM, Borges AP, Sato DMV, et al. (2016) Automatic knowledge learning using Case-Based Reasoning: A case study approach to automatic train conduction. *International Joint Conference on Neural Networks (IJCNN)* 4579–4585.

43 Zhao H, Chen H, Dong W, et al. (2017) Fault diagnosis of rail turnout system based on case-based reasoning with compound distance methods. *29th Chinese Control And Decision Conference (CCDC)* 4205–4210.

44 Hadj-Mabrouk H (2017) Preliminary Hazard Analysis (PHA): New hybrid approach to railway risk analysis. *International Refereed Journal of Engineering and Science* 6: 51–58.

45 Hadj-Mabrouk H (2016) Machine learning from experience feedback on accidents in transport. *7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications* 246–251.

46 Ganascia JG (1987) Agape et Charade: deux mécanismes d'apprentissage symbolique appliqués à la construction de bases de connaissances. Thèse d'État, Université Paris-sud, France.

47 Quinlan JR (1986) Induction of Decision Trees. *Mach Learn* 1: 81–106.

48 Hadj-Mabrouk H (2016) CLASCA: Learning System for Classification and Capitalization of Accident Scenarios of Railway. *Journal of Engineering Research and Application* 6: 91–98.

49 Hadj-Mabrouk H (2018) A Hybrid Approach for the Prevention of Railway Accidents Based on Artificial Intelligence. *International Conference on Intelligent Computing & Optimization* 383–394.

50 Hadj-Mabrouk H (2019) Contribution of artificial intelligence to risk assessment of railway accidents. *Journal of Urban Rail Transit* 5: 104–122.

51 Hadj-Mabrouk H, Mejri H (2015) ACASYA: a knowledge-based system for aid in the storage, classification, assessment and generation of accident scenarios. Application to the safety of rail transport systems. *Advances in Computer Science an International Journal* 4: 7–13.

52 Hadj-Mabrouk H (2017) Contribution of learning Charade system of rules for the prevention of rail accidents. *Intell Decis Technol* 11: 477–485.

53 Aamodt A, Plaza E (1994) Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Commun* 7: 39–52.

54 Harmon P (1991) Case-based reasoning II. *Intelligent Software Strategies* 7: 1–9.

55 Kolodner J (1992) An introduction to case-based reasoning. *Artif Intell Rev* 6: 3–34.

56 Leake D (1996) CBR in Context: The present and future. *Case-Based Reasoning: Experiences, Lessons, and Future Directions* 3–30.

57 Mott S (1993) Case-based reasoning: Market, applications, and fit with other technologies. *Expert Syst Appl* 6: 97–104.

58 Pinson S, Demourioux M, Laasri B, et al. (1993) Le Raisonnement à Partir de Cas: panorama et modélisation dynamique. Séminaire CBR, LAFORIA, Rapport 93/42, 1er octobre.

59 Slade S (1991) Case-based reasoning: A research paradigm. *AI Mag* 12: 42–55.

60 Hadj-Mabrouk H (2017) Case-Based Reasoning for the Evaluation of Safety Critical Software. Application to The Safety of Railway Transport. *International Journal of Engineering Research and Development* 13: 37–43.

61 Hadj-Mabrouk H (2019) Contribution of artificial intelligence and machine learning to the assessment of the safety of critical software used in railway transport. *AIMS Electronics and Electrical Engineering* 3: 33–70.

62 Shannon CE (1948) A mathematical theory of communication. *Bell Syst Tech J* 27: 379–423.