Urban Rail Transit (2019) 5(2):104–122
https://doi.org/10.1007/s40864-019-0102-3 http://www.urt.cn/

ORIGINAL RESEARCH PAPERS

Contribution of Artificial Intelligence to Risk Assessment of Railway Accidents

Habib Hadj-Mabrouk[1]

Received: 9 February 2019 / Revised: 1 May 2019 / Accepted: 6 May 2019 / Published online: 31 May 2019

Abstract In the design, development, and operation of a rail transport system, all the actors involved use one or more safety methods to identify hazardous situations, the causes of hazards, potential accidents, and the severity of the consequences that would result. The main objective is to justify and ensure that the design architecture of the transportation system is safe and presents no particular risk to users or the environment. As part of this process of certification, domain experts are responsible for reviewing the safety of the system, and are being brought in to imagine new scenarios of potential accidents to ensure the exhaustiveness of such safety studies. One of the difficulties in this process is to determine abnormal scenarios that could lead to a particular potential accident. This is the fundamental point that motivated the present work, whose objective is to develop tools to assist certification experts in their crucial task of analyzing and evaluating railway safety. However, the type of reasoning (inductive, deductive, by analogy, etc.) used by certification experts as well as the very nature of the knowledge manipulated in this certification process (symbolic, subjective, evolutionary, empirical, etc.) justify that conventional computer solutions cannot be adopted; the use of artificial intelligence (AI) methods and techniques helps to understand the problem of safety analysis and certification of high-risk systems such as guided rail transport systems. To help experts in this complex process of evaluating safety studies, we decided to use AI techniques and in particular machine learning to systematize, streamline, and strengthen conventional approaches used for safety analysis and certification.

Keywords Rail safety · Risk assessment · Accident scenarios · Artificial intelligence · Machine learning · Expert system

1 Introduction

This article describes a contribution for improving the usual safety analysis methods applied in the certification of railway transport systems. The methodology is based on complementary and simultaneous use of knowledge acquisition and machine learning. We used the ACASYA software environment to support the safety analysis aid methodology. ACASYA aims to provide experts with suggestions of potential failures which have not been considered by the manufacturer and which may jeopardize the safety of a new rail transport system. In more formal terms, the methodology used to assist in such safety analysis is based on two models: a generic accident scenario representation model, which is based on a static and a dynamic description of a scenario, and a model of the implicit reasoning of the expert, which involves three major activities, namely the classification, evaluation, and generation of scenarios.

The first level (CLASCA) relates to finding the class to which a new scenario which has been suggested by the manufacturer belongs. The second level (EVALSCA) is used to compare the list of summarized failures (SFs) suggested by a manufacturer with a list of stored historical SFs in order to stimulate the formulation of hazardous

✉ Habib Hadj-Mabrouk
habib.hadj-mabrouk@ifsttar.fr

[1] French Institute of Science and Technology for Transport, Development and Networks, 14-20 Boulevard Newton, 77447 Marne-la-Vallée, France

Communicated by Xuesong Zhou.

situations not 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. These two levels are supplemented by a third level (GENESCA), which makes use of the static and the dynamic description of the scenario (the Petri model). The generation of a new scenario is based on injecting an SF, defined in the previous level as being plausible, into a specific sequencing of the change in the marking of the Petri net.

Considering the scale of the problem, the design and construction of this demonstration model of the ACASYA system concentrated on the first two levels of processing (classification and evaluation of scenarios). After presenting the objectives of and motivations for this study, the following paragraphs successively present the approach adopted for the development of the system of assistance for the analysis and evaluation of rail safety as well as the results obtained.

2 Main Methods of Railway Safety Analysis

The notion of "risk" is a combination of the probability of the occurrence of a potential accident and the severity of the most severe damage that it could cause. This is usually expressed on a scale with several levels of risk: "negligible," "unacceptable," "tolerable," "acceptable under certain conditions," and "acceptable." This "level of risk" can also be measured by the number of probable deaths per predefined time unit. There is also the notion of "safety," defined by the European standard CENELEC 50129 [1] as the absence of any risk with unacceptable level, measured on a qualitative scale.

In terms of railway safety, there are two major safety activities (Fig. 1). The first is usually called the development or "construction of safety" process, while the second activity focuses on managing safety (coordination, organization, etc.). The safety development process can, in turn, be hierarchically structured into four safety analysis activities: (1) system-level analysis, (2) automation-level analysis, (3) hardware-level analysis, and (4) software-level analysis.

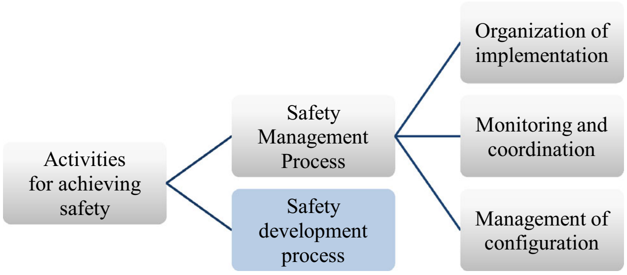

Activities for achieving safety
Safety Management Process
Safety development process
Organization of implementation
Monitoring and coordination
Management of configuration

Fig. 1 Activities for achieving safety

2.1 The Different Levels of Safety Analysis

Each level of safety analysis has one or more safety methods (Fig. 2):

- At the system level, the main method is the "preliminary hazard analysis" (PHA) method, which 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 [2].
- At the level of automation, a method known as "functional safety analysis" (FSA) is applied. FSA aims to justify that the design architecture of the system is safe against potential accidents identified by the PHA and thereby ensure that all safety provisions are taken into account to cover potential hazards or accidents.
- At the software level, several methods related to software safety analysis (SSA) are carried out. SSA is generally based on the software errors and effects analysis (SEEA) [3] method as well as on critical code reads.

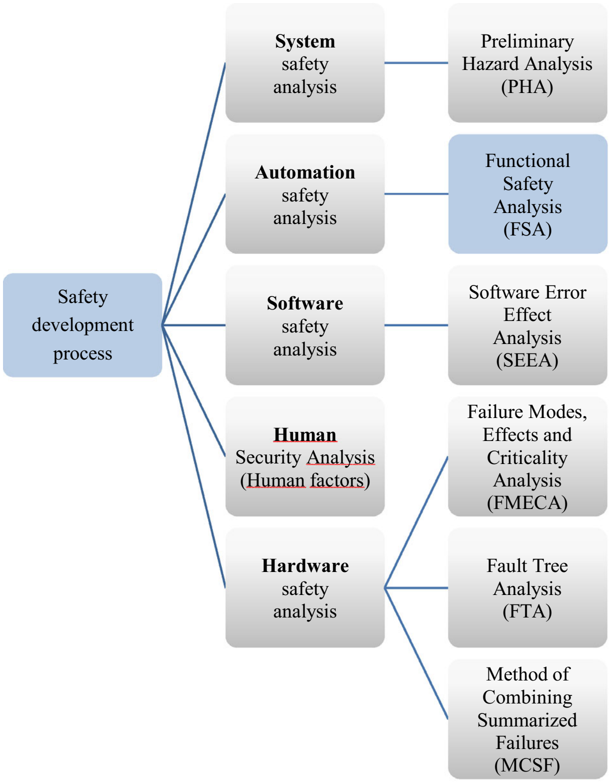

Safety development process
System safety analysis
Automation safety analysis
Software safety analysis
Human Security Analysis (Human factors)
Hardware safety analysis
Preliminary Hazard Analysis (PHA)
Functional Safety Analysis (FSA)
Software Error Effect Analysis (SEEA)
Failure Modes, Effects and Criticality Analysis (FMECA)
Fault Tree Analysis (FTA)
Method of Combining Summarized Failures (MCSF)

Fig. 2 Main methods of analysis of rail transport safety [5]

- At the hardware level, several safety methods relating to hardware safety analysis (HSA) need to be established. HSA focuses on safety analysis of electronic boards and interfaces.

In this approach, two types of analysis are implemented: inductive and deductive.

2.2 Inductive and Deductive Analyses

- *Inductive* analysis is based on analysis of failure modes, their effects, and their criticality (AFMEC). The AFMEC method is usually completed by the method of combining summarized failures (MCSF) [4] also called the significant failures combination search method. Originating 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 safety analysis of Concorde and Airbus planes. The AFMEC method, which usually highlights simple failures, must be supplemented by studying combinations of failures that would result in undesirable (or dangerous) events. Thus, the MCSF method, as extended in the AFMEC method, inductively determines such combinations of failures. Generally, it is noted that the effects (or consequences) on the system are identical for one or more modes of failure. These failure modes are then grouped into fault sets called "summary faults" (SFs). This method of safety analysis therefore focuses on extracting only combinations of security-relevant failures and can be considered 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 failures (SFs).
- *Deductive* analysis proceeds by searching for scenarios that run counter to safety and that make it impossible to comply with the safety criteria derived from functional safety analysis (FSA). Such analysis usually requires the use of the cause tree method.

Indeed, all of these methods of safety analysis are based on two fundamental approaches, of inductive or deductive type. In the inductive approach, the reasoning goes from the most particular to the most general, leading to a detailed study of the effects of a failure on the system and its environment. In other words, inductive methods start from elementary events, either to look for consequences directly or to identify combinations of events that may have other than minor consequences. PHA, AFMEC, and SEEA are examples of such inductive methods. In the deductive process, the reasoning goes from the most general to the most particular such that, in the face of a failing system, the causes of the failure are deduced. The main deductive method is fault tree analysis (FTA). In practice, and when considering a complex system, safety analysis requires experts in the field to implement an iterative safety development process involving both inductive and deductive methods. In the design, development, and operation of a rail transport system, all the actors involved (infrastructure manager, railway companies, manufacturer, certification body, national safety authority, investigative technical body, etc.) use one or more safety methods to identify hazardous elements and equipment, hazardous situations, the causes of hazards, potential accidents, and the severity of the consequences that would result.

Having briefly presented the main methods of safety analysis involved in the design and development of a rail transport system, the following paragraphs are devoted to the design and implementation of a functional safety analysis (FSA) support tool (Fig. 2) based on artificial intelligence techniques and in particular on automatic learning methods. The next paragraph details the motivations for and objectives of this study.

3 Objectives of and Motivations for the Study

The ACASYA project, which is the subject of this paper, provides assistance particularly for the functional safety analysis (FSA) phase. FSA aims to justify that the design architecture of a system is safe against potential accidents identified by PHA and thereby ensure that all safety provisions are taken into account to cover potential hazards or accidents. Such analyses 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. FSA can highlight unsafe scenarios that require improvement and rectification of the specification anddesign phases of the transport system.

The development of this FSA support tool was motivated by various findings revealed by the problem identification and specification phase, the main ones being:

- The need to rationalize and automate the classical FSA approach
- The need to improve the quality of accident risk analysis by archiving, formalizing, and disseminating the knowhow of the builder and safety experts
- The difficulty of synthesizing and exploiting the considerable amount of historical knowledge involved in FSAs carried out for guided rail transport systems that have already been certified and put into service in France

- The desire to judge the comprehensiveness of FSAs proposed by the manufacturer as part of the review of the safety studies of a new rail transport system

These reasons guided us towards the development of an AI-based tool for assisting in the analysis and review of the completeness and consistency of the new FSA. More precisely, this tool must:

- Suggest risks that were not taken into account during the initial analysis.
- Help find the most appropriate solutions or preventive measures to guard against a particular risk.
- Propose a common risk analysis database for the FSA for the various actors involved in the development of a rail transport system. Such a database is indispensable, especially when the studied system bears a resemblance to existing systems but experience is lacking. It can be further enriched and updated by the various stakeholders. The main objective of this database is to store the experience and knowhow in the analysis of risks and errors related to FSA.

Considering the complexity of the knowledge of experts and the difficulty which they have in explaining their mental processes, there is a danger that the extracted knowledge will be incorrect, incomplete, or even inconsistent. A variety of research in AI is in progress in an attempt to understand this problem of expertise transfer. Research is currently taking place in two major, independent areas [6]:

- *Knowledge acquisition*, which aims to define methods for achieving a better grasp of expertise transfer. These methods chiefly involve software engineering and cognitive psychology.
- *Machine learning*, which involves the use of inductive, deductive, abductive, or analogical techniques to provide the knowledge-based system (KBS) with learning.

To develop a KBS to aid in safety analysis, we combined these two approaches and used them in a complementary way.

4 Approach Adopted for Evaluation of Railway Safety

The approach taken to design and implement the tool for assisting with safety analysis and assessment is based on two main activities [7]. The first 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 the “historical scenarios knowledge base” (HSKB). The second activity aims to exploit this stored historical knowledge (HSKB) to develop safety analysis knowhow that can help experts judge the comprehensiveness of safety analyses. This second activity is essentially based on the use of machine-learning techniques and in particular on inductive learning of production rules. More precisely, the solutions chosen to design and implement this tool involve the following four main steps:

- Formalization and structuring of knowledge with a view to identifying a model for the representation and acquisition of FSA accident scenarios based on manufacturers’ safety files.
- Collection of knowledge and more precisely risks and solutions adopted. In this step, we exploit the formalism elaborated in the previous step to archive typical cases.
- Creation of a knowledge base covering all FSAs for already certified transport systems.
- Exploitation of the database to help judge the completeness of an FSA of a new system.

5 Literature Review in Artificial Intelligence

5.1 Introduction to Artificial Intelligence

The ambitious aim of artificial intelligence is to equip computers with some of the faculties of the human mind, viz. to learn, recognize, reason, etc. The ability to understand natural language and reason is the keystone of AI. Excitement about AI and machine learning is now ubiquitous, with a hundred new conferences appearing all over the world during 2019 [3], including AI and the Future of Work in San Francisco, the Global Symposium on AI in China, the Summit on In-Depth Financial Learning in London, AI and Big Data again in London, the AI Conference in New York, the International Conference on Machine Learning in California, the Summit on AI in Hong Kong, the Summit on Deep Learning in Montreal, Canada, etc.

The first results of research carried out in this field concerned expert systems or KBS, which have emerged as decision-support tools capable of replicating some of the intellectual tasks usually performed by human experts. The capacity to exploit and especially capitalize on and sustain such experience gives KBS the power of information and decision to guide nonspecialist users. Since the 1970s, KBS have made a remarkable entry into industry; they are no longer considered rare objects from research laboratories. However, they very rarely achieve the performance of the human expert, and are often poorly adapted to the real needs of end-users. This is due to the difficulty of extracting the necessary expertise from one or more experts in the field and representing this knowledge without distortion to construct a cognitive model of the expert. In addition, the manual filing of the KBS knowledge base (KB) is a crucial phase. Indeed, capturing knowledge to store it in the KB of an expert system is a complex and time-consuming task and often requires great material and human resources. Several research works have evoked this problem of collecting and formalizing the knowledge manipulated by the problem-solving expert. Experts may have great difficulty in explicitly describing the steps they use in reasoning to make decisions, which may require a long process of reflection to enable the unconscious part of the approach to be explained. However, the success of a KBS project depends on this difficult and sometimes painful task. Like many authors, we consider this task as the bottleneck in KBS development. Indeed, given the complexity of the knowledge of the expert and the difficulty of the latter in explaining their mental processes, the knowledge extracted may often be inaccurate, incomplete, or even incoherent. Various research studies in AI have thus been carried out to address this problem of expertise transfer, divided into two major independent research activities [8–15].

In *knowledge acquisition* (KA), the design of a knowledge base requires the extraction, analysis, structuring, and formalization of the knowhow of a domain that is accessed through one or more qualified experts. Therefore, the transfer of this expertise raises the following delicate questions: Who really holds the expertise? How can it be accessed? How can it be extracted? How can it be formalized without distortion? Which representation should be chosen? How can the collected knowledge be validated and maintained? Various research studies have been conducted to improve the understanding of these problems inherent to knowledge acquisition and KBS design. Methods, techniques, and tools for knowledge acquisition are now accessible to the cognitive engineer (knowledge engineer) and to the expert, offering a methodological framework for KBS development. Possible techniques for knowledge extraction have been studied and presented by many authors [8–10].

Machine learning (ML) [12, 13] is an important branch of research in the field of AI. The birth of this discipline dates back to the 1960s, and the most spectacular result was obtained at that time by the American A. Samuel, who designed a program to play checkers that memorized many moves and constantly improved its strategy, eventually reaching champion level. The principle of learning was born: to learn is to perfect one's knowledge and improve one's performance by taking advantage of past failures. In the 1970s, a new approach to learning emerged: AI, which aims for explicability in the knowledge base formed. Thus, apprenticeship, which at first was only an interesting idea, has now become an indispensable discipline for the progress of several industrial systems, being regarded as a promising solution to assist knowledge acquisition. Work in this area [11] has attempted to answer certain issues, such as how such a mass of knowledge can be expressed clearly, managed, added to, and modified. Machine learning is defined by a dual objective: a scientific objective (understanding and mechanically producing phenomena of temporal change and the adaptation of reasoning) and a practical objective (automatic acquisition of knowledge bases from examples). Learning may be defined as performance improvement through experience, being intimately connected to generalization [12, 13]; learning consists of making the transition from a succession of experienced situations to knowledge which can be reutilized in similar situations. Three types of problems are raised for each of the main learning techniques [11]. The first of these is grouping (termed "classification" in data analysis), i.e., given a certain mass of knowledge, how is it possible to discover links between the different items in order to group them into meaningful and simpler subgroups? The second problem (discrimination) is that of learning classification procedures: given a set of examples of concepts, how is it possible to find a method which provides effective recognition of each concept? The third problem is that of generalization: how is it possible, on the basis of concrete examples of a situation, to find a formula which is sufficiently general to describe the situation in question, and how is it possible to explain the descriptive ability of this formula? ML has attracted increasing interest in recent years, as evidenced by the impressive number of publications and conferences on this subject. ML's efforts to address clustering, discrimination, and generalization of objects have resulted in a wide variety of methods, techniques, algorithms, and systems. Nevertheless, this abundant literature makes it difficult to comprehend the field, given the ambiguity of its vocabulary and the absence of rigorous reference definitions.

In addition to KBS based on knowledge acquisition (KA) and machine learning (ML) as mentioned above, artificial intelligence implements several other methods and techniques, such as neural networks (NN) also called deep learning, genetic algorithms (GAs), pattern recognition, which is often associated with image processing, fuzzy systems based on the fuzzy set theory proposed by Lotfi Zadeh in 1965, big data analytics (BDA), and case-based reasoning (CBR). For those involved or interested in the latest AI technologies, the annual Applied Artificial Intelligence Conference is a beneficial event, focusing on the latest trends in and future impact of AI (Artificial Intelligence) applications and commercialization in many sectors, including transportation, logistics, health, energy, financial technologies, the future of work (FoW), the Internet of Things (IoT), and cybersecurity.

5.2 Examples of Applications of Artificial Intelligence in Railway Transport

In recent years, researchers and experts in the field of land and air transport have become increasingly interested in the application of artificial intelligence techniques to solve certain decision assistance problems, such as diagnosis of transport equipment, management of maintenance operations, analysis of driver behavior, prediction of deterioration of transport infrastructure, 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 [16], (fuzzy knowledge-based) expert systems for rail traffic control [17], deep learning for detection of lateral railroad defects [18], and neural networks for detection of defects on the surface of rails [19]. Meanwhile, big data analytics (BDA) has been used in particular to identify trends, discover relationships, implement predictive analysis, and give meaning to images, data flows, and various other types of information. In railway transport applications, BDA can make a beneficial contribution in view of the large amounts of data generated in the transport system by sensors installed on tracks or wagons, or signaling equipment, monitoring and inspection equipment, communication systems, train monitoring systems, etc. BDA can examine the collected dataset to obtain useful information to explain, for example, the potential causes of degradation during operation, failure of certain track components, and possibly safety equipment. BDA thus presents the main characteristics required by rail transport experts to monitor the overall condition of the infrastructure, optimize and plan maintenance operations, manage the risks of accidents and potential incidents, and consequently improve the safety of the transport system. As an example, one can mention the work on exploiting data relating to operation, maintenance, and railway safety [20], decision-making on rail maintenance [21], engineering and management of railway applications [22], improvement of call reporting systems [23], implementation of a predictive approach to the safety and maintenance of personnel [24], and the work of Siemens on the use of big data to build the Internet of Trains [25].

5.3 Contributions with Respect to the State of the Art

As stated above, 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 the considerable historical knowledge of functional safety analysis (FSA) and the desire to judge the completeness of an FSA proposed by the manufacturer during the development of a new rail transport system. This need to rationalize the traditional FSA approach to improve the quality of accident risk analyses and finally assist experts in judging the completeness of an FSA and the adequacy of the protective measures considered directed us towards the development of a tool allowing the suggestion of potential accidents and/or the most appropriate protective or preventative measures for a particular risk. To achieve these objectives, the chosen approach is based on several aspects of artificial intelligence and in particular on the use of the following techniques:

- Knowledge acquisition to gather knowledge on railway safety and in particular the scenarios of potential accidents.
- Learning by classification of concepts to group accident scenarios into homogeneous classes, e.g., relating to train collisions or derailment problems. In addition, learning by classification to help certification experts in the search phase of accident scenarios or potential incidents likely to jeopardize safety studies proposed by the manufacturer of the rail transport system.
- Rule-based machine learning (RBML) to automatically identify, from a base of historical scenarios (experience feedback), relevant safety rules, which are often difficult to extract manually from safety experts.
- A knowledge-based system (KBS) to which production rules, previously deduced by machine learning, are transferred to construct the knowledge base for the FSA assessment support tool.

Thus, our rail safety assessment approach is a hybrid method built around a classification algorithm, a rule-based automatic learning system (RBML), and a system based on the knowledge (KBS). Despite the undeniable interest in artificial intelligence approaches as presented in the previous paragraph, there is no comprehensive approach to meet all of our research objectives and needs for analysis and railway safety assessment.

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 the definition of the expected performance of the learning system. For the safety problem and the certification of rail transport systems, the knowledge acquisition phase identified some 80 accident scenarios relating to the risk of collision. This set of scenarios was grouped by the security expert into nine classes of scenario, such as the redundancy switching class and the initialization class. These scenario classes are archived in the HSKB. This database of learning examples is not completely representative of the field of railway safety and is tainted with "noisy" data.

The objective of this study is to apply machine learning on this basis 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 into 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 a hazardous situation contrary to security), the learning mechanism must produce a base of rules of the form: "if symptoms then failures", exploitable by the inference engine of an expert system.

Having briefly recalled 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 assist with analysis of rail safety and to the current proposal as perceived through the literature review. The properties imposed on the rail safety analysis tool are as follows:

- *Similarity-based learning* (SBL), characterized by the availability of a large number of examples supplying the learning system but a lack of knowledge on the field (or weak knowledge). Given the acquired safety knowledge, which is essentially accident scenarios, the choice of the SBL is justified.
- *Symbolic-digital data processing*. This approach combines the efficiency of digital processing that allows operation in the presence of noisy and incomplete security data, and the explicability of symbolic processing necessary for the user to understand the knowledge produced.
- *Classification learning and empirical regularity learning*. Two learning strategies are required to achieve 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*. The classification activity requires nonmonotonic incremental learning of conjunctive descriptions of accident scenario classes. The scenario evaluation activity requires the generation of production rules to assist in recognition of failures and dangers.
- *Rules structuring*. The learning system must generate, rather than isolated rules with a single inference step, a system of structured rules that enables deductive reasoning, essentially taking into account the orientation of the rules, viz. from symptoms to causes (failures).
- *Nonmonotonic learning*. Incrementality is an indispensable property when dealing with the evolving nature of knowledge in the field of railway safety. It must be nonmonotonic to ensure the possible questioning of previously learned knowledge. To guarantee such nonmonotony of knowledge, it is necessary to integrate means that allow its stabilization and thus 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 control of the learned knowledge. The system, meanwhile, must argue its reasoning and decisions. This "interactive" or "supervised" learning promotes the acquisition of new knowledge. Such interaction 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 of the accident scenarios, one could consider using the CHARADE [12] system for the generation of production rules, although we are forced to develop a new classification system for accident scenarios.

5.3.1 Rationale for Choosing the CHARADE Rule Learning System

CHARADE [12] allows not only the generation of a structured rule system that can be exploited by the inference engine of an expert system, but also completion of the description of the examples provided by the expert to take into account possibly noisy data, such as example accident scenarios. This makes it possible to learn certain logical rules and uncertain rules simultaneously, modulated by a likelihood coefficient. Finally, its major originality lies in its flexibility and its translation of the desired KBS functionalities thanks to the constraints that it implements. All of these benefits come at the cost of learning, which cannot be incremental. However, at the level of the development of evaluation rules for accident scenarios, the structuring of the rules has priority over incrementality.

5.3.2 *Need to Develop a New Classification Learning System: CLASCA*

Analysis of 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 [26] 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 in this 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 with a prominent place for the expert to judge, semantically, the quality of the knowledge produced is an interesting advance. Indeed, an apprenticeship supervised by an 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 preclassified by the expert and not representative of the field, without obliging the expert to list all the examples but by involving them 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 scenario provided by the expert to incrementally form conjunctive descriptions of potential scenario classes that are comprehensible by the expert and compatible with the CHARADE system. This approach, which lies somewhere between nonincremental 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 that is conceived and detailed below.

The preceding paragraphs 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 for the analysis and evaluation of functional safety. This is essentially based on the joint use of the CHARADE and CLASCA modules previously identified.

6 General Description of the Safety Assessment Methodology

This article describes a contribution to improving the usual safety analysis methods applied in the certification of railway transport systems. The methodology is based on the complementary and simultaneous use of knowledge acquisition and machine learning. We use the ACASYA software environment to support the safety analysis assisting methodology. As shown in Fig. 3, ACASYA consists of four main modules. The formalization module deals with the acquisition and representation of a scenario and is part of the knowledge acquisition phase. The three other modules (CLASCA, EVALSCA, and GENESCA) deal with the problems associated with scenario classification, evaluation, and generation.

7 Acquisition of Safety Knowledge

The knowledge acquisition phase emerges as a critical step in identifying issues related to the safety and certification of rail transportation systems. It allows identification of the main aspects as well as the human and technical environment and finally to frame the project in the following two regards: (1) a feasibility study of a system to support the analysis of functional security and (2) to focus primarily on the risk of "collision." The extraction and formalization of all accident or incident scenarios is a significant task. In fact, the safety of rail transport requires all the risks of potential accidents (collision, derailment, electrocution, etc.) to be taken into account, to which many scenarios can be associated. We deliberately limit this feasibility study to one risk, viz. "collision." Nevertheless, the architecture of the realized system is open and could therefore accommodate other risks.

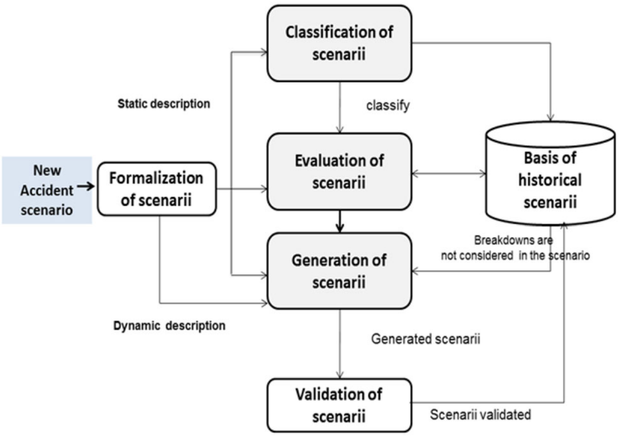


Fig. 3 Functional organization of the ACASYA system

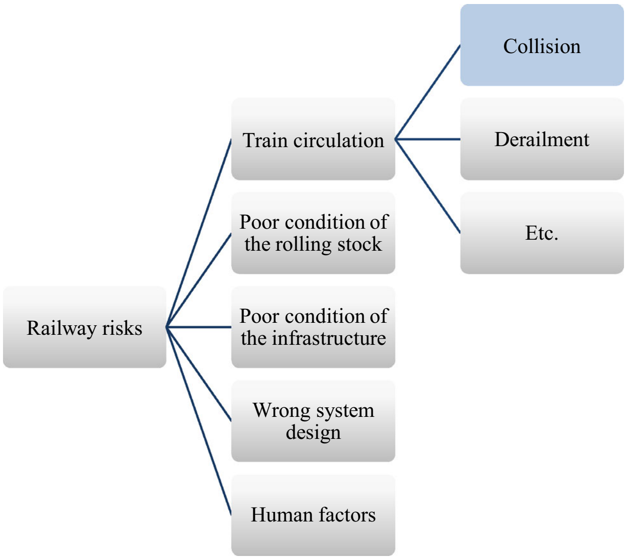


Fig. 4 The main causes of railway risks

As shown in Fig. 4, the main causes of rail risk are generally related to train circulation, poor condition of rolling stock, poor condition of infrastructure, incorrect system design, or human factors. Each cause of a railway accident, can be, in turn, broken down into several sub-causes; For example, the risks associated with train movements can lead to the risk of collision or train derailment. We deliberately limited this feasibility study of the safety assessment support tool to the risk of "collision." Knowledge on the "collision" risk is structured into a hierarchy (Fig. 5). We distinguish three collision classes: collision with operating trains, collision with fixed obstacles, and collision with other vehicles.

To date, the knowledge acquisition phase has resulted in the development of a historical scenarios knowledge base (HSKB) that includes 80 scenarios of accidents or incidents related to collision risk. Figure 6 shows an excerpt from the list of accident scenarios, 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, and crossing a breakpoint in manual driving. All 80 scenarios were subsequently grouped by safety experts into several classes or families of scenarios (Fig. 7) such as the class of "redundancy switching," the class of "initialization sequence," the class of "location of trains," or the class of "emergency braking management."

This HSKB, which forms the basis of the learning examples, is exploited by the CLASCA learning algorithm to determine the membership class of a new scenario proposed by the transport system manufacturer. This HSKB database is also exploited by the CHARADE learning algorithm to produce rules necessary for learning the summary fault (SF) recognition functions. This work is presented below.

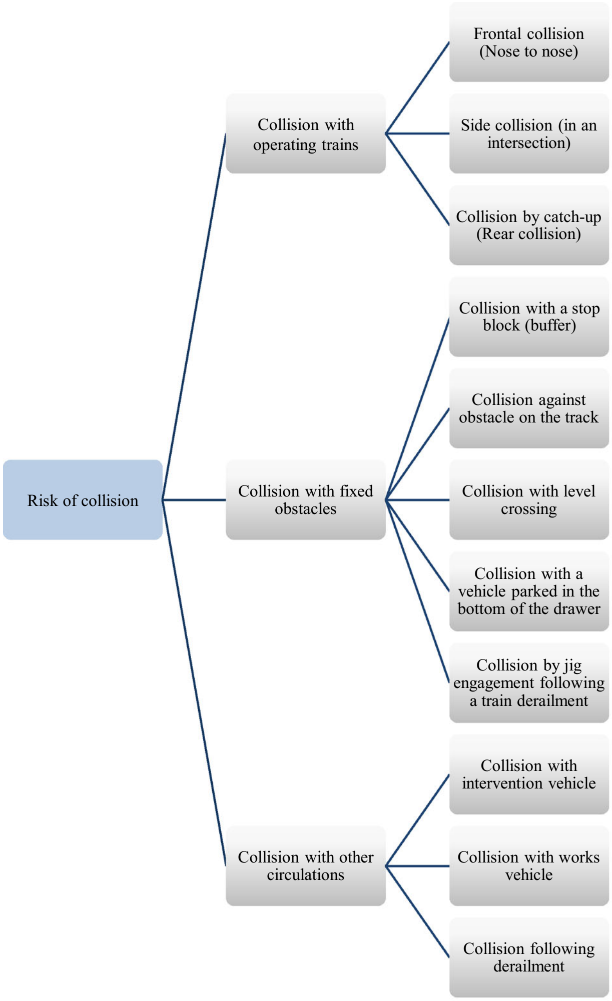

Risk of collision
Collision with operating trains
Collision with fixed obstacles
Collision with other circulations
Frontal collision (Nose to nose)
Side collision (in an intersection)
Collision by catch-up (Rear collision)
Collision with a stop block (buffer)
Collision against obstacle on the track
Collision with level crossing
Collision with a vehicle parked in the bottom of the drawer
Collision by jig engagement following a train derailment
Collision with intervention vehicle
Collision with works vehicle
Collision following derailment

Fig. 5 Classification of different types of "collision" risk

8 Modeling of Accident Scenarios

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, whereas the second is "dynamic" and reveals 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. On the other hand, the dynamic description (modeled by a Petri net) is implemented in the framework of the

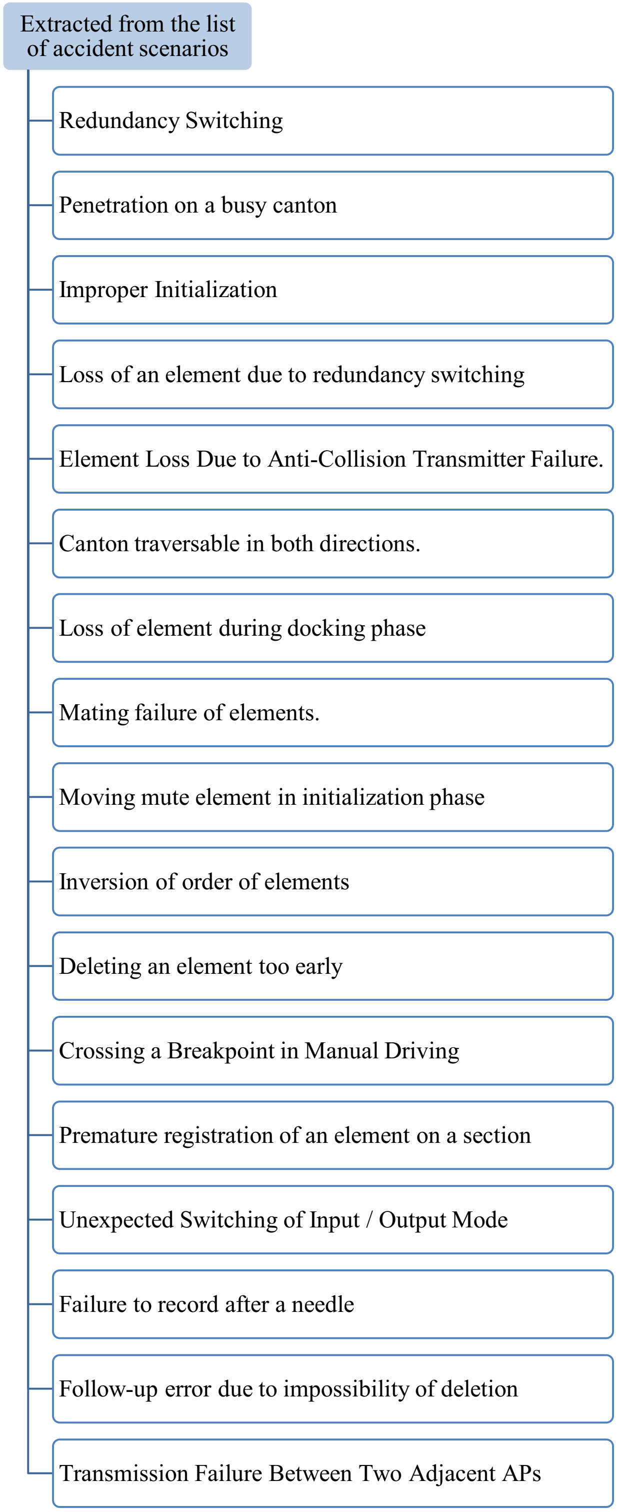

Extracted from the list of accident scenarios
Redundancy Switching
Penetration on a busy canton
Improper Initialization
Loss of an element due to redundancy switching
Element Loss Due to Anti-Collision Transmitter Failure.
Canton traversable in both directions.
Loss of element during docking phase
Mating failure of elements.
Moving mute element in initialization phase
Inversion of order of elements
Deleting an element too early
Crossing a Breakpoint in Manual Driving
Premature registration of an element on a section
Unexpected Switching of Input / Output Mode
Failure to record after a needle
Follow-up error due to impossibility of deletion
Transmission Failure Between Two Adjacent APs

Fig. 6 A sample of scenarios acquired from domain experts

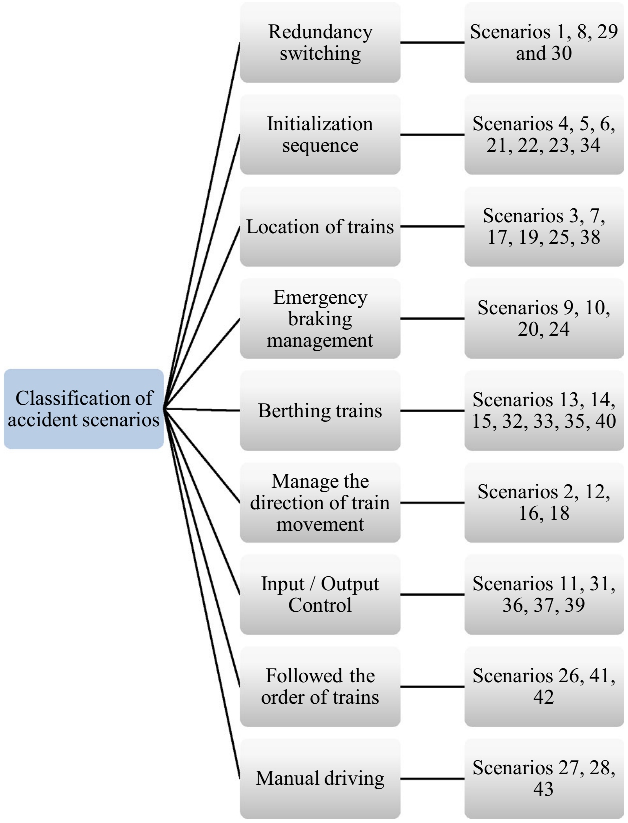

Classification of accident scenarios
Redundancy switching
Scenarios 1, 8, 29 and 30
Initialization sequence
Scenarios 4, 5, 6, 21, 22, 23, 34
Location of trains
Scenarios 3, 7, 17, 19, 25, 38
Emergency braking management
Scenarios 9, 10, 20, 24
Berthing trains
Scenarios 13, 14, 15, 32, 33, 35, 40
Manage the direction of train movement
Scenarios 2, 12, 16, 18
Input / Output Control
Scenarios 11, 31, 36, 37, 39
Followed the order of trains
Scenarios 26, 41, 42
Manual driving
Scenarios 27, 28, 43

Fig. 7 Grouping of scenarios into nine classes by the safety expert

third, GENESCA module dedicated to the generation of new accident scenarios. The study presented herein focuses solely on the static description. 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 (Fig. 8; Table 1), and each

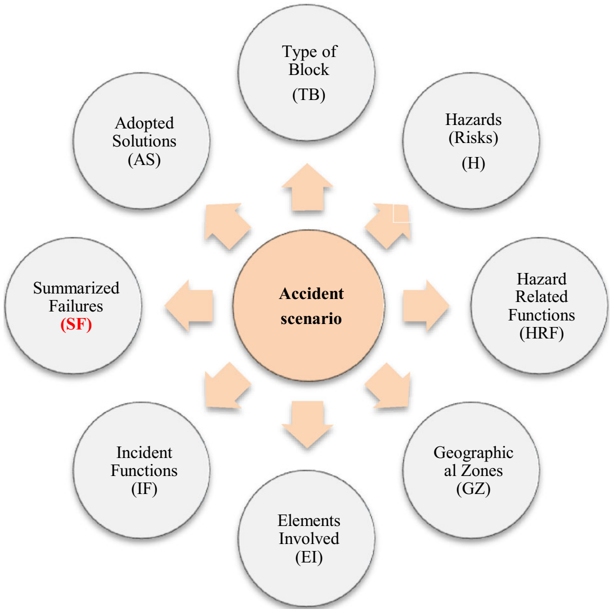


Fig. 8 Parameters describing an accident scenario

Table 1 Definition of the descriptive parameters of an accident scenario

Parameter	Definition
Type of block	In guided rail transport systems, there are two types of block: fixed blocks and moving blocks
Hazard	Hazard measures the level of risk. It is characterized by two factors: the severity of consequences and the probability of occurrence. In the present context, the term "hazard" is used to describe a situation which may give rise to a physical injury or the destruction of one or more pieces of equipment of the transport system. Several potential hazards have been identified, e.g., collision, derailment, and electrocution
Hazard-related functions	These are protective functions which are intended to remove the hazard or make it acceptable to the user. A specific hazard may involve several subfunctions of the system, of either a protective or purely functional nature; For example, redundancy switching, pushing, and train localization functions are relate to the collision hazard
Geographical zones	The five main geographic zones are as follows: terminus, station, line, train entry zone, and section limit
Elements involved	It was possible to make a list of those involved in all the accident scenarios, for example, the trains, the operator at the control center (CC), and the train driver
Incident functions	These are functions which are related to the operation of the system and which can promote the occurrence of a scenario which affects the safety of the system. These functions may play the role of a catalyst. Four basic incident functions have been identified: route management, traffic control, transmission, and operating instructions (consistency of instructions and operator vigilance)
Summarized failures (SF)	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
Adopted solutions	The manufacturer suggests several solutions for each safety-threatening scenario, one of which is implemented. A list of adopted solutions (ASs) for the scenarios which have been collected so far has been drawn up

attribute is associated with a list of possible values (Table 2). This "descriptive form" was subsequently used as the basic form for the acquisition of the 80 scenarios. In summary, the static description of a scenario leads to the definition of a first descriptive language for the example scenarios. This is a classical representation by attribute–value couples. This expression language is close to the language of the expert and has the advantage of being compatible with the structuring of historical safety data. The scenarios which have been collected so far in the historical knowledge base relate to the collision problem and have been constructed on the basis of the safety dossiers of French rail transport systems [VAL, POMA 2000, MAGGALY, and TVM430 (Nord TGV)] and the knowhow of experts. More precisely, the level of detail which is required in the system description to formalize the scenarios is essentially related to the general specifications of the system, the functional specifications, and the functional safety analysis.

9 Detailed Description of Safety Assessment Methodology

As illustrated in Fig. 9, the railway safety analysis and assessment methodology is organized into the 11 steps, detailed below. The first eight steps are carried out by the scenario classification module (CLASCA) and the last three in the scenario evaluation module (EVALSCA):

1. Acquisition of the scenarios to be treated
2. Construction of the historical scenarios knowledge base (HSKB)
3. Predesign: parameters and learning constraints
4. Learning: introduction of description of classes of scenarios
5. Classification of a new example of a scenario
6. Validation of knowledge learned by the system
7. Study of convergence of the learning system
8. Update of the HSKB database
9. Learning the summarized failures (SF) recognition functions: evaluation of the rule base produced by CHARADE
10. Deduction of SFs to be considered in the manufacturer's scenario
11. Validation by the safety expert

9.1 Description of the CLASCA Classification Learning System

The first level of analysis relates to finding the class to which a new scenario which has been suggested by the manufacturer belongs. The aim of 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 situations which have been described by the manufacturer and certain experienced or envisaged situations involving equipment which

Table 2 List of descriptive parameters of an accident scenario

Attribute	Possible values
Type of block (TB)	Fixed blocks
	Moving blocks
Hazards (risks) (H)	Collision
	Derailment
	Electrocution
	Fire or explosion
	Passenger pinch, cut on any object
	Passenger struck during closing of doors
	Poorly controlled emergency evacuation
Hazard-related functions (HRFs)	Management of automatic driving
	Train localization
	Control input/output
	Tracking trains
	Speed set point
	Management of train stops
	Authorization CI/HT
	Redundancy switching
	Initialization
	Manual driving
	Alarm management
	Route protection
	Traction/braking
	Evacuation
Incidental functions (IFs)	Route management
	Traffic control
	Communication (transmission)
	Instructions (consistency, vigilance)
Elements involved (EIs)	Number of trains
	Mobile operator
	Operator at the control center (CC)
	Automatic driver (AD) with redundancy
	Automatic driver (AD) without redundancy
Geographical zones (GZs)	Terminus
	Station
	Line
	Train entry zone
	Section limit
Summarized failures (SFs)	Element and target in opposite direction
	Train reversing into an occupied block
	Collision avoidance transmitter failure
	Masking of an alarm by initialization
	Unexpected route switching by AP
	Invisible element on the driving zone
	Location fault
	Authorization of both directions of travel for the same train
	Change of meaning of the target
	Element and target in the opposite direction
	Unexpected I/O mode switching
	Penetration on a section by recoil
	Incorrect operation of door opening

Table 2 continued

Attribute	Possible values
	Incorrect start of the train
	Untimely reactivation of an electric section
	Incorrect points movement
Adopted solutions (ASs)	Prohibit change of route if the approach area is occupied
	Increase the length of the Canton
	The upstream Canton must be released only if the downstream is occupied
	One-way authorization at a time
	Take into account the distance of recoil
	Prohibit switching of I/O mode while a mode is in progress
	Empty the section of any element before initializing
	Take into account the position of the train when switching I/O mode
	Prohibit switching of I/O mode while a mode is in progress
	Impose a reduced speed for formations of more than two elements
	Immobilize the train until the evacuation alarm is acknowledged
	Control of the absence of needle movement by the AP
	In MC, respect the minimum spacing from the preceding train

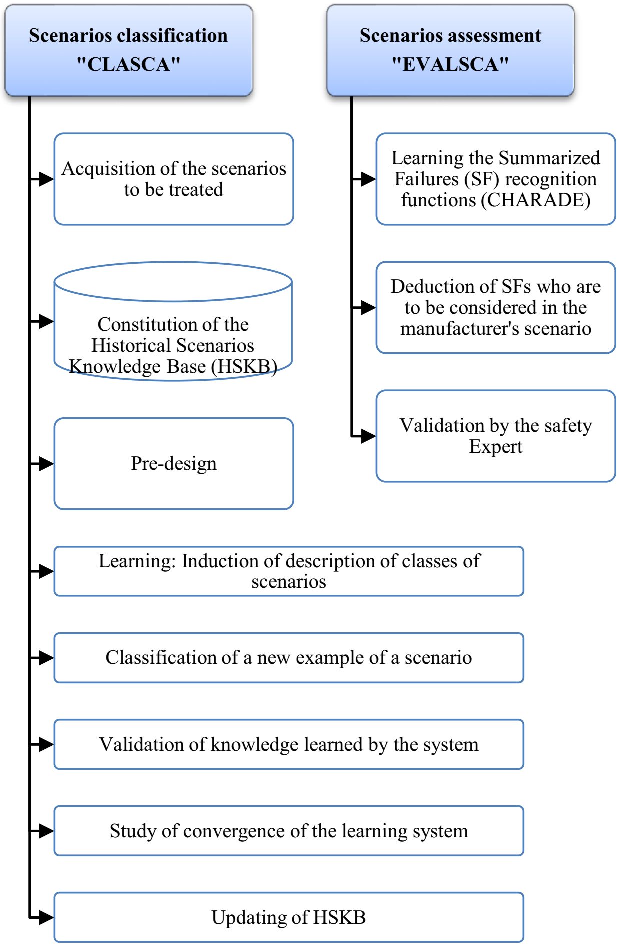

Scenarios classification "CLASCA"
Acquisition of the scenarios to be treated
Constitution of the Historical Scenarios Knowledge Base (HSKB)
Pre-design
Learning: Induction of description of classes of scenarios
Classification of a new example of a scenario
Validation of knowledge learned by the system
Study of convergence of the learning system
Updating of HSKB
Scenarios assessment "EVALSCA"
Learning the Summarized Failures (SF) recognition functions (CHARADE)
Deduction of SFs who are to be considered in the manufacturer's scenario
Validation by the safety Expert

Fig. 9 Safety analysis and assessment methodology

has already been certified and approved. The classification of a new scenario includes the following two major phases (Fig. 10):

- A characterization (or generalization) stage for constructing a description of each class of scenarios. This stage operates by detecting similarities within a set of historical scenarios in the HSKB which have been preclassified by the expert in the domain.
- A deduction (or classification) stage to determine the class to which a new scenario belongs by evaluating a similarity criterion. The descriptors of the new scenario (static description) are compared with the descriptions of the classes which were generated previously.

This initial level of processing not only provides assistance to the expert by suggesting scenarios which are

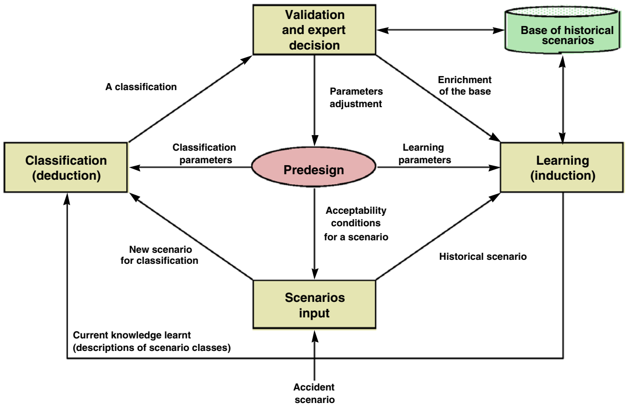

Validation and expert decision
Base of historical scenarios
A classification
Parameters adjustment
Enrichment of the base
Classification (deduction)
Classification parameters
Predesign
Learning parameters
Learning (induction)
Acceptability conditions for a scenario
New scenario for classification
Historical scenario
Scenarios input
Current knowledge learnt (descriptions of scenario classes)
Accident scenario

Fig. 10 General architecture of CLASCA module

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. CLASCA is a learning system based on research into classification procedures. It is inductive, incremental, and dedicated to the classification of accident scenarios. Learning in CLASCA is on the one hand nonmonotonic 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 that produced by the system. CLASCA incrementally develops conjunctive descriptions of historical scenario classes in order to characterize a set of insecure situations and identify a new scenario submitted for evaluation to experts. The classification operation is carried out in the seven consecutive steps detailed below.

9.1.1 Acquisition of the Scenarios to be Treated

This first step concerns the capture of three types of scenarios: (1) a new scenario (to verify its admissibility before processing), (2) a historical scenario (a preclassified scenario experienced by the experts, archived and with known class), and (3) a scenario to be classified (a new scenario to be classified, whose consistency the expert seeks to evaluate). The scenarios analyzed to date are related to the risk of “collision.” They come not only from the experience and knowhow of railway safety experts, but also from experience feedback obtained from guided rail transport systems such as the VAL system, the POMA system, the MAGGALY system, and the TVM430 system TGV-North.

9.1.2 Construction of the Historical Scenarios Knowledge Base (HSKB)

This first stage involves the collection of safety analysis knowledge with respect to guided rail transport systems. This knowledge is as follows:

- The HSKB, which currently includes about 80 historical scenarios related to a collision risk. These scenarios have been formalized on the basis of a static description, then grouped into classes by the expert.
- An accident scenario description language, which consists of a set of descriptors (or parameters which describe a scenario).
- Accident scenarios, described using this language. These may be historical and preclassified by the expert in order to add to the HSKB, or new and suggested by the manufacturer. In the second case, the experts will attempt to evaluate the consistency of the scenarios.
- Learning parameters (induction, classification, and convergence parameters).

Very schematically, guided rail transport systems are considered as being an assembly of basic bricks, thus a new system possesses certain bricks which are shared by systems which are already known. In the context of this study, the basic bricks which have currently been identified have been grouped together, and the ACASYA tool finds and then exploits shared bricks in order to deduce the class to which a new scenario belongs or evaluate its completeness.

9.1.3 Predesign

This step makes it possible to set the different values of the parameters and learning constraints required by the system. During this phase, the user defines in particular the learning parameters (induction, classification, and convergence parameters) and the admissibility constraints of a scenario, which define the conditions necessary for a scenario to be acceptable by the system. All of these parameters mainly affect the relevance and quality of the classification knowledge learned, as well as the speed of convergence of the system.

9.1.4 Description of Classes of Scenarios

This stage involves generalizing the classes which have been predefined by the experts in order to generate a comprehensive description for each class which both characterizes the division conducted by the expert and enables identification of the class to which the new example belongs. 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) in order to limit the loss of information (formula no. 1). The description of a class is further enriched by taking into account the associated summarized failures (SF) which are involved. These SFs will subsequently be exploited in order to develop the base of learning examples. The objective is to determine the frequency of occurrence τ in a selected example class C_k of attribute A of rank n.

$\tau_m^n(C_k)$ denotes the probability that attribute A_n takes the value V_m^n in example $\boldsymbol{e_p^K}$ and corresponds to the occurrence frequency of the value in class C_k:

$$\tau_m^n(C_k) = \frac{\sum_{p=1}^{\mathrm{Card}\,C_k} D_m^n\left(e_p^k\right)}{\mathrm{Card}\,C_k}. \tag{1}$$

9.1.5 Classification of a New Example of a Scenario

In this stage, a new example of a scenario is assigned to an existing class C_k. To achieve this, it is necessary to define a classification criterion which measures the degree of resemblance between the new example and each of the preexisting classes. This similarity criterion is based on statistical calculations and takes account of the semantics of the domain of application. The classification phase of a new example accident scenario requires the definition of a classification parameter called the "adequacy rate" (T_{ad}), which measures the degree of resemblance between the new example E_i and each of the classes C_k of preexisting scenarios. T_{ad} is expressed as

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

This 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 the list of descriptors relevant to characterize each class of examples from the set of descriptors identified with the experts. 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 approach makes it possible to define a second rate of adequacy, which reflects the semantics of knowledge:

$$T_{\mathrm{ad}\,\mathbf{2}}(E_i, C_k) = \frac{\sum_{(m,n)/(A_n, V_m^n)\,\mathrm{soit}\,''\mathrm{cl\acute{e}}''\,\mathrm{de}\,C_k} D_m^n(E_i) \times \tau_m^n(C_k)}{\sum_{(m,n)/(A_n, V_m^n)\,\mathrm{soit}\,''\mathrm{cl\acute{e}}''\,\mathrm{de}\,C_k} \tau_m^n(C_k)} \tag{3}$$

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 the semantics of the data:

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

with $0 \le \lambda \le 1$.

λ is a smoothing coefficient that can be adjusted experimentally or proposed by the domain expert to take account of his deep convictions. It is possible to give more or less importance to statistical or semantic processing; For example, if $\lambda = 0$, the matching rate is purely semantic, whereas if $\lambda = 1$, it is purely statistical, while the two types of treatment are taken into account equally in the case where $\lambda = 0.5$.

9.1.6 Validation of Knowledge Learned by the System

The probable membership class of the new scenario identified must be validated by the expert using a dialog module which enables argumentation by the system and a decision by the expert. In argumentation, the system keeps track of the deduction phase to build its explanation. Following this phase of justification of the classification decision, the expert decides either to accept the proposed classification, in which case the scenario will be learned by the classification algorithm, or to reject the classification. In the second case, it is up to the expert to decide what to do next; For example, they can adjust the learning parameters (λ, ss, etc.), create a new class, modify the description of the scenario, or put the scenario on hold. In the situation where the tool assigns the new example of a scenario to a class, this class needs to be updated. The updating process generates four situations as below:

- Particularization of descriptors, viz. descriptors which are considered characteristic at instant t may lose their significance at instant ($t + 1$)
- Generalization of descriptors: descriptors considered to be not meaningful may become characteristic
- Simultaneous particularization and generalization
- Learning of new descriptors which enrich the description of the class

This phenomenon demonstrates the nonmonotonic character of the learning system.

9.1.7 Study of Convergence of the Learning System

The integration of a new example into a class causes a refresh of the frequency of appearance of the descriptors. In this context, the unavoidable presence of "noise" makes nonmonotonic 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 decided to change the value of the similarity threshold ss throughout the classification cycle, to become increasing "demanding" with growth of the size of the considered class. This approach leads to the definition of two types of convergence: "internal convergence," related to the stability of knowledge within a class, and "global convergence," which 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):

$$ss(C_k, n) = \left(1 - \alpha\,\mathrm{e}^{\beta(1-\mathrm{Card}\,C_k)}\right)\left(1 - \gamma\mathrm{e}^{\delta(n_0-n)}\right), \tag{5}$$

where

- $ss(C_k,\ n)$ is the similarity threshold that increases monotonically as a function of Card C_k and n and tends to 1. It is updated with each addition of an example to a class.

- Card C_k is the cardinality of class C_k at time t, which evolves with the addition of new examples to a class C_k.
- n_0 is the set of learning examples provided to the system from the start.
- n is the number of total examples that cover the domain considered.
- α, β, γ, and δ are four coefficients calculated from four initial conditions to be determined. Note that the values of α, β, γ, and δ ($\alpha > 0$, $\beta > 0$, $\gamma > 0$, $\delta > 0$) can be set differently from one class to another. β and δ affect the learning time and consequently the speed of convergence. These two factors affect the convergence. Variation of β and δ allows the user to evolve the system at will and ensure its convergence.

9.1.8 Update of HSKB Database

A scenario that is classified by the system, judged to be relevant, and validated by the expert will subsequently be integrated into the HSKB database. This phase includes updating the data and therefore learning new scenarios of potential accidents.

9.2 Description of the Expert System to Aid in Evaluation of Safety Based on Learning of Rules: EVALSCA

The second level of processing (EVALSCA) considers the class to which CLASCA has deduced that the scenario belongs, in order to evaluate the consistency of the manufacturer's scenario. The evaluation approach is centered on the summarized failures (SFs) which are involved in the manufacturer's scenario. An accident scenario describes a set of circumstances that can lead to a dangerous situation. It is characterized by a context and a set of parameters, in particular an SF, risk (hazard or potential accident), the actors involved, the incidental functions, and a 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 of the SFs involved in all the scenarios collected so far has been compiled. The following list is a sample of a few SFs:

- SF1: train reversing into an occupied block
- SF2: collision avoidance transmitter failure in a train
- SF3: masking of an alarm by initialization

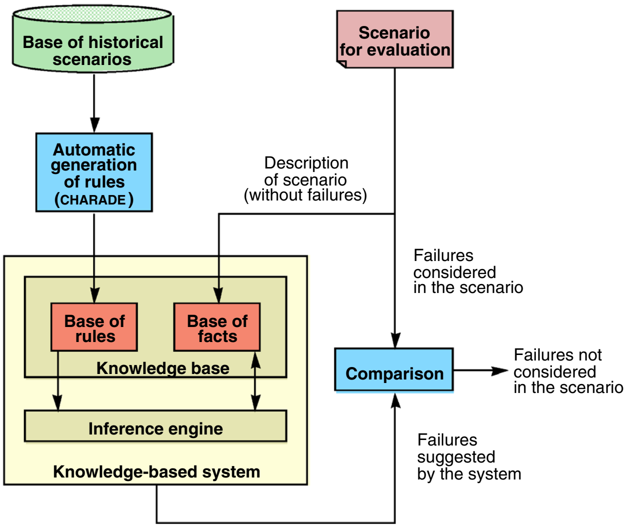
Consequently, the evaluation of a scenario involves the following two modules (Fig. 11):

- 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.

9.2.1 Learning the SF Recognition Functions

The goal is to generate a recognition function for each SF associated with a given 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. This involves a logical dependence, which can be expressed as shown in Fig. 12.

In this way, a base of evaluation rules can be generated for each class of scenarios. This phase of learning attempts to generate a system of rules by using the base of 80 examples which was formed previously. A base of evaluation rules can be generated for each class of scenarios. The conclusion of each generated rule should contain the SF descriptor or fact. In this context, it has proved indispensable 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 literature led us to choose the CHARADE mechanism [12]. The ability of CHARADE to automatically generate 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



Base of historical scenarios
Scenario for evaluation
Automatic generation of rules (CHARADE)
Description of scenario (without failures)
Failures considered in the scenario
Base of rules
Base of facts
Knowledge base
Inference engine
Knowledge-based system
Comparison
Failures not considered in the scenario
Failures suggested by the system

Fig. 11 General architecture of EVALSCA module

```
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)
```

Fig. 12 Form of SF recognition rules

CHARADE is based on looking for and discovering empirical regularities which are present in the entire learning sample. A 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. To illustrate this rule generation principle, let us assume that there is a learning set which consists of three examples E1, E2, and E3:

- E1 = d1 & d2 & d3 & d4
- E2 = d1 & d2 & d4 & d5
- E3 = d1 & d2 & d3 & d4 & d6

In this case, CHARADE can detect an empirical regularity between the combination of descriptors (d1 & d2) and the descriptor d4. All those examples described by d1 & d2 are also described by d4. The rule d1 & d2 → d4 is thus obtained (Fig. 13).

9.2.2 Deduction of SFs To Be Considered in the Manufacturer's Scenario

The SF deduction stage requires a preliminary phase during which the rules generated by CHARADE are transferred to an expert system to construct a scenario evaluation knowledge base. The purpose of the EVALSCA module is to compare the list of SFs suggested in a manufacturer scenario with the list of stored historical SFs (in the rule base of the expert system), so as 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. This may thus promote the generation of new accident scenarios.

Assuming that the constructor's scenario consists of the following facts: "moving_block," "collision," "management_of_automatic_driving-train_monitoring," "initialization," "terminus," "operator_at_CC," "ad_without_redundancy," and "instructions," the inference engine of the expert system, proceeding by forward chaining of the rules generated by CHARADE, can deduce the summary failure (SF) shown in Fig. 14.

```
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]

If      Type of block = fixed block,
        Hazard related functions = initialization
        Incident functions = instructions
Then    summarized failures = SF10
        (Erroneous re-establishment of safety frequency/high voltage,)
        Hazard related functions = Full control/High voltage permission
        Hazard related functions = alarm management,
        Hazard related functions = train localization.
        [0]

If      Hazard related functions = train localization,
        Elements involved = AD without redundancy.
Then    summarized failures = SF9
        (Entry of a train into an occupied block),
        Geographical zone = line,
        Type of block = fixed block.
        [0]
```

Fig. 13 A sample of some rules generated by CHARADE

```
@@ 09/02/2019
     moving_block
     collision
     management_of_automatic_driving-train_monitoring
     initialization
     terminus
     operator_at_CC
     ad_without_redundancy
     instructions
DEDUCTION:
     Summarized failure = SF19 (Silent train)
```

Fig. 14 Example result of deduction by the expert system

10 Perspectives: Towards the Development of a System to Aid Automatic Generation of Accident Scenarios (GENESCA Module)

The two levels of processing described above make use of the static description of the scenario (descriptive parameters). They are supplemented by a third level (GENESCA) which makes use of the "dynamic description" of the scenario (the Petri model) and three reasoning mechanisms, namely induction, deduction, and abduction. Generation of a new scenario is based on injecting an SF, defined by the previous level as plausible, into a specific sequencing of the change in the marking of the Petri net (Fig. 15). In view of the scale of the problem, the design and construction of the demonstration model of the ACASYA system concentrated on the first two levels of processing (classification and evaluation of scenarios).

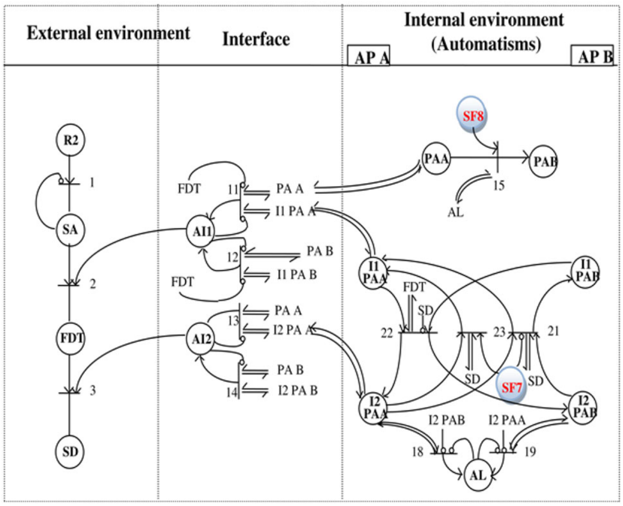


Fig. 15 Example of modeling an accident scenario by Petri net

11 Conclusions

The essence of the task of analyzing rail safety is to imagine new scenarios of potential accidents that either demonstrate the completeness of the safety analysis performed by the manufacturer or contradict it. Indeed, in the presence of a dangerous situation described in the form of scenarios by the manufacturer, the expert reasons by analogy, that is, trying to bring a new situation of insecurity closer to certain situations experienced with equipment or similar systems that have already been certified. Before making an adequate decision, the expert looks for similarities between the submitted case and a set of typical simulated or experienced cases with which they have already been confronted. In this sense, improving the quality of the expert's decisions by implementing a tool to help find similarities between different configurations or situations of insecurity represents a substantial aid for safety experts. This paper presents our contribution to the improvement of the methods which are normally used to analyze and assess the safety of automatic devices in guided transport systems. This contribution is based on the use of artificial intelligence techniques and has involved the development of several approaches and tools to assist in the modeling, storage, and assessment of knowledge about safety. The software tools have two main purposes: firstly to record and store experience concerning safety analyses, and secondly to assist those involved in the development and assessment of the systems in the demanding task of evaluating safety studies. Currently, these tools are at the mock-up stage, but a first validation by security experts has shown the interest of the suggested approaches. This study aims to complete and improve the model of representation of accident scenarios by taking into account human factors [27].

Acknowledgements I want to thank Mr. Xuesong Zhou (Associate Executive Editor-in-Chief), the Reviewers and Springer Correction Team for the advice they gave me, the confidence they showed me and the care that they brought to the correction of this manuscript.

References

1. CENELEC—EN 50129 (2003) Railway applications—communication, signaling and processing systems—safety related electronic systems for signaling
2. Hadj-Mabrouk H (2017) Preliminary hazard analysis (PHA): new hybrid approach to railway risk analysis. Int Refereed J Eng Sci 6(2):51–58
3. 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 Electron Electr Eng 3(1):33–70. https://doi.org/10.3934/ElectrEng.2019.1.33
4. Villemeur A (1988) Sûreté de fonctionnement des systèmes industriels, Paris, Eyrolles, coll. «Collection de la direction des études et recherches d'Électricité de France», juillet 1988, ISSN: 0399-4198
5. Hadj-Mabrouk H (2016) Machine learning from experience feedback on accidents in transport. In: 7th International conference on sciences of electronics, technologies of information and telecommunications, 18–20 Dec. 2016, https://doi.org/10.1109/setit.2016.7939874, pp 246–251, http://ieeexplore.ieee.org/document/7939874/
6. Hadj-Mabrouk H (2017) Contribution of learning Charade system of rules for the prevention of rail accidents. J Intell Decis Technol 11(4):477–485. https://doi.org/10.3233/idt-170304
7. Hadj-Mabrouk H (2019) A hybrid approach for the prevention of railway accidents based on artificial intelligence. In: P. Vasant et al (eds) Intelligent computing & optimization (ICO 2018). Chapter: 41, advances in intelligent systems and computing (AISC), vol 866. Springer, Berlin, pp 383–394. https://doi.org/10.1007/978-3-030-00979-3_41
8. Aussenac G, Gandon F (2013) From the knowledge acquisition bottleneck to the knowledge acquisition overflow: a brief French history of knowledge acquisition. Int J Hum Comput Stud 71(2):157–165
9. Gaines BR (2012) Knowledge acquisition: past, present, and future. Int J Hum Comput Stud 71(2):135–156. https://doi.org/10.1016/j.ijhcs.2012.10.010
10. Dieng R (1990) Méthodes et outils d'acquisition des connaissances, ERGO IA90, Biarritz, France, 19 à 21 septembre
11. Kodratoff Y (1986) Leçons d'apprentissage symbolique automatique. Cepadues éd, Toulouse, France
12. Ganascia J-G (1987) Agape et Charade: deux mécanismes d'apprentissage symbolique appliqués à la construction de bases de connaissances. Thèse d'État, Université Paris-sud, France
13. Ganascia J-G (2011) Logical induction, machine learning and human creativity. In: Switching codes. University of Chicago Press, ISBN 978022603830
14. Michalski R-S, Wojtusiak J (2012) Reasoning with missing, not-applicable and irrelevant meta-values in concept learning and pattern discovery. J Intell Inf Syst 39(1):141–166

15. Jamal S, Goyal S, Grover A, Shanker A (2018) Machine learning: What, why, and how? In: Shanker A (eds) Bioinformatics: sequences, structures, phylogeny. Springer, Singapore. https://doi.org/10.1007/978-981-13-1562-6_16
16. Bergmeir C, Sáinz G, Martínez Bertrand C, Benítez JM (2013) A study on the use of machine learning methods for incidence prediction in high-speed train tracks. In: Ali M, Bosse T, Hindriks KV, Hoogendoorn M, Jonker CM, Treur J (eds) Recent trends in applied artificial intelligence (IEA/AIE 2013). Lecture notes in computer science, vol 7906. Springer, Berlin
17. Fay A (2000) A fuzzy knowledge-based system for railway traffic control. Eng Appl Artif Intell 13(6):719–729. https://doi.org/10.1016/S0952-1976(00)00027-0
18. Santur Y, Karaköse M, Akin E (2017) A new rail inspection method based on deep learning using laser cameras. In: International artificial intelligence and data processing symposium (IDAP). https://doi.org/10.1109/idap.2017.8090245
19. Faghih-Roohi S, Hajizadeh S, Núñez A, Babuska R, De Schutter B (2016) Deep convolutional neural networks for detection of rail surface defects. In: International joint conference on neural networks (IJCNN), 24–29 July 2016, Canada. https://doi.org/10.1109/ijcnn.2016.7727522
20. Ghofrania F, He Q, Goverde R, Liud X (2018) Recent applications of big data analytics in railway transportation systems: a survey. Transp Res C Emerg Technol 90:226–246. https://doi.org/10.1016/j.trc.2018.03.010
21. Thaduri A, Galar D, Kumar U (2015) Railway assets: a potential domain for big data analytics. Procedia Comput Sci 53:457–467. https://doi.org/10.1016/j.procs.2015.07.323
22. Nii Attoh-Okine (2014) Big data challenges in railway engineering. In: IEEE international conference on big data (big data), Washington, DC, USA. https://doi.org/10.1109/bigdata.2014.7004424
23. Hughes P (2018) Making the railway safer with big data, 30.01.18. http://www.railtechnologymagazine.com/Comment/making-the-railway-safer-with-big-data
24. Hayward V (2018) Big data & the digital railway. https://on-trac.co.uk/big-data-digital-railway
25. Marr B (2017) How Siemens is using big data and IoT to build the internet of trains, May 30, 2017. https://www.forbes.com/sites/bernardmarr/2017/05/30/how-siemens-is-using-big-data-and-iot-to-build-the-internet-of-trains/#2b7a4b6e72b8
26. Quinlan JR (1986) Induction of decision trees. Mach Learn 1:81–106
27. Hadj-Mabrouk H (2018) New approach of assessing human errors in railways. Trans VSB Tech Univ Ostrava Saf Eng Ser 13(2):1–17. https://doi.org/10.2478/tvsbses-2018-0007