# Identification of human errors and influencing factors: a machine learning approach

Caroline Morais[a,b], Ka Lai Yung[c], Karl Johnson[d], Raphael Moura[a,b], Michael Beer[a,e,f], Edoardo Patelli[a,d,*]

[a] *Institute for Risk and Uncertainty, University of Liverpool, Chadwick Building, Peach Street, Liverpool L69 7ZF, United Kingdom*
[b] *National Agency for Petroleum, Natural Gas and Biofuels (ANP), Av. Rio Branco, 65, CEP 20090-004, Centro, Rio de Janeiro, RJ, Brazil*
[c] *Faculty of Applied Science & Engineering, University of Toronto 35 St. George Street, Room 157, Toronto, ON M5S 1A4, Canada*
[d] *Centre for Intelligent Infrastructure, University of Strathclyde, James Weir Building, 75 Montrose St, Glasgow G1 1XJ, United Kingdom*
[e] *Institute for Risk and Reliability, Leibniz Universität Hannover, Callinstr. 34, 30167 Hannover, Germany*
[f] *Tongji University, Shangai, China*

## Abstract

The capability of learning from accidents from different industrial sectors could prevent similar accidents to happen. With this aim, the Multi-attribute Technological Accidents Dataset (MATA-D) has been created, using a classification focused on the relation between human errors and their influencing factors (e.g., cognitive functions, organisational and technological factors). The process of collecting new data for this dataset should be constant, not only to decrease epistemic uncertainty in human reliability data but also to reflect changes in human behaviour due to evolving technology and organisational arrangements.

However, reading an accident report is a time-consuming process, which delays the learning process. For this reason, this research proposes an automated approach to train the computer on a predefined classification scheme (taxonomy), which will be called the *virtual human factors classifier*. The virtual classifier should support human experts to analyse accident reports for organizational, technological, and individual factors that may trigger human errors.

The proposed approach is based on classifying text according to previously labelled accident reports by human experts. Two case studies are used to demonstrate how data from different sectors can be used to train the machine, providing an efficient cross-discipline knowledge transfer. The accuracy of the results is promising and comparable to the classifications provided by human experts. The proposed work demonstrated to the industry the feasibility of the use of artificial intelligence to collect data and support risk and reliability assessments, and recommendations based on the study findings are suggested for investigation agencies.

*Keywords: automated text classification, accident report data, human factors taxonomy, human reliability data, CREAM*

## 1. Introduction

One of the most acknowledged ways to prevent design errors in complex industries is to conduct risk assessment, where multi-disciplinary teams revise a design according to information from past accidents, components, and human reliability. There are industrial recommended practices on how companies should use lessons learnt from past accidents (CCPS, 2010), research on how they are actually using it (Drupsteen et al., 2013) or how it could be used (Moura et al., 2017a; Moura et al., 2017b). The lessons learnt encompass not only hazards but also their frequency of occurrence, which are used to quantify risks in probabilistic risk analysis, or to estimate order of magnitude in semi-quantitative analysis (e.g. LOPA) and qualitative analysis when risk ranking is required (Baybutt, 2016).

Regarding frequency, component failure databases play a central role in quantitative risk analysis, where data is majorly provided by components manufacturers and sometimes shared within groups of industry operators, such as the Maintenance Steering Group (MSG-3) in aviation (EASA; Gonçalves and Trabasso, 2018) and the Offshore and Onshore Reliability Data (OREDA) in upstream oil & gas (Lima et al., 2019; OREDA). However, there is still plenty of space for the development of databases to support *system safety*, which should be able to include systems and installations rather than only components' parts, as well as the interaction between human, organizational and technological factors (Leveson, 2020).

[*] Corresponding author at: edoardo.patelli@strath.ac.uk; Centre for Intelligent Infrastructure, University of Strathclyde, James Weir Building, 75 Montrose St, Glasgow G1 1XJ, United Kingdom

To fill this information gap, a human reliability database has been created comprising major accidents from different industry sectors (with the same level of complexity), all classified with an established human reliability taxonomy (Moura et al., 2016). The database, known as MATA-D, has currently 238 accident events classified into 53 variables, including human erroneous actions and their influencing factors (Moura et al., 2020). Although it is already possible to use it for human reliability analysis (Morais et al., 2021 (in press); Morais et al., 2020), it would be desirable to reduce its uncertainty, leading to more precise risk estimates. To understand how to decrease its uncertainty, it is important to understand the different representation of the uncertainties within the dataset: aleatoric to model uncontrollable events, e.g. impairments and cognitive bias, or epistemic/reducible uncertainty due to missing data and theoretically reducible (Patelli et al., 2016). It is acknowledged in the human reliability field that human behaviour is dependent on the context, varying according to organizational and technological factors (Hollnagel, 1998). The lack of information on these factors' interactions (seldomly observed and reported) is the major contribution to the epistemic uncertainty. Thus, to reduce epistemic uncertainty it would be desirable to expand the database, by collecting more accident reports and classifying them in order to increase the chance of describing more human-machine-organisation interactions.

However, collecting empirical data is time-consuming and expensive, especially in human reliability field, where data collection and classification are usually done by other humans (experts in their fields). MATA-D database have been constructed through extensive reading and classifying 238 accident investigation reports (Moura et al., 2016), a task that have taken around one year to be completed. The classification also required specialised knowledge, as the assessors had to be minimally trained on the taxonomy used to pursue the classification.

The present study proposes to enlarge a human reliability dataset by replacing (or supporting) human coding by automated classification of accident reports from any industrial sector using a pre-defined human factor's taxonomy. In order to absorb lessons learnt from different industry sectors, the objective is to continually add to the dataset reports only from industries with the same level of complexity regarding the interaction of organisational structure, technology and humans (Moura et al., 2016). The work hereby presented is a substantial improvement and extension of the strategy proposed by some of the authors of this paper in a conference (Morais et al., 2019). Therefore, the aim of the present research is not only to expand MATA-D, but to do it faster and timely. The use of a machine-learning strategy for text recognition and classification is herein proposed because an experienced expert takes around 3 days to read and classify one accident report, which contains about two hundred pages. A machine-learning algorithm takes less than one minute. Thus, it would be interesting to develop a computer support, that could support risk specialists, or directly collect and update the database for every new accident report of interest. Caution would be needed on the acceptance criteria of this new data, as depending on the sample quality the uncertainty might increase (Siegrist, 2011). Therefore, a central research question of this study is whether a machine learning approach is capable of both accelerating the expansion of a human reliability database and maintaining the same data quality offered by human experts.

The approach, here called as *virtual human factors classifier* might be useful in other ways. For instance, it may be used to improve human reliability Bayesian and credal networks (Morais et al., 2021 (in press); Morais et al., 2020), or to support cross-learning from different industry sectors. It can also support incident investigators in an unbiased fashion to consider possible performance shaping factors, which might have triggered human errors (instead of focusing only on human errors). On the original aim of expanding MATA-D, risk assessors should benefit for the provision of more data thus more possible combinations between performance shaping factors and human errors, minimising missing data problem in the use of data for probabilistic approaches.

This paper has been divided into four parts. The first part gives a brief overview of the recent history of major accident data. The second section of this paper will examine the options of machine-learning strategies and performance metrics. The third section is concerned with the dataset, taxonomy and the methodology used for this study. The fourth section presents the findings of the research, focusing on the case study of including the analysis of two accident reports from aviation (Boeing 737 MAX) and oil & gas industry (FPSO CDSM, Cidade de Sao Mateus floating production storage and offloading unit).

## 2. Theoretical background

This section explores the literature regarding previous similar research regarding the investigation of accidents in different industry sectors, the selection of the most used machine-learning algorithms, and most appropriate performance metrics.

### *2.1. Related work in similar industry sectors*

The present research has focused on previous studies that have used machine-learning strategies to classify textual narratives into safety and risk features. The sample also focused in industries with similar level of organisational and technological complexity as found in MATA-D, as well as those that have investigated at least one human factor as one of the features, such as aviation (Robinson et al., 2015), railway (Heidarysafa et al., 2018; Hughes et al., 2017), oil & gas (Ribeiro et al., 2020), civil construction (Goh and Ubeynarayana, 2017) and maritime industries (Grech et al., 2002). A comprehensive review of the application of machine-learning techniques in occupational accident analysis, however, mixing many industries with lower level of complexity is provided in (Sarkar and Maiti, 2020).

Despite large research and application of machine-learning approaches, gaps and needs for risk and reliability analysis remains. Previous studies have not classified full accident reports into a human reliability taxonomy – nor any attempts have been identified to expand databases of human reliability with the support of machine-learning, or within multiple industry sectors. For instance, only one specific human factor (situation awareness) has been analysed in maritime accident reports (Grech et al., 2002) while often the aim was to analyse near-misses or close call reports (daily basis reports that consist of small narratives from workers (Hughes et al., 2017), to support safety managers on having timely decisions upon risk controls (Goh and Ubeynarayana, 2017; Heidarysafa et al., 2018; Ribeiro et al., 2020; Robinson et al., 2015).

The highest performance obtained are from the studies with texts sizes of around 200 words, and which have collapsed many classes into a few more frequent ones. However, the need to expand the MATA-D to support better risk analysis is to classify full major accident reports (with text sizes of around 200 pages) and to not discard nor collapse classes that are less labelled (sparse data).

### *2.2. Human-categorized text*

Readers can easily categorize a document into its topic if they have the classification scheme in mind, an action that can be described as *manual coding* (Grech et al., 2002) and *human-categorization* (Goldberg, 2017). In cases where more than one *coder* or *rater* classifies the same documents, it is good practice to measure the interrater agreement with a coefficient, such as Cohen's kappa (Kim et al., 2020).

Although human categorization is considered the standard approach, it is time-consuming and resource demanding. It is also prone to error, in particular when involving large databases (Robinson et al., 2015). The manual assessment of accident reports has been used by Moura et al. to create the Mata-D, after reading 238 accident reports and classifying them as Boolean values according to factors described in Table 1 (0 if a factor was not reported, 1 if a factor was reported), as represented in Figure 1. A step-by-step description of how the information has been classified is shown in (Moura et al., 2016) and the resulting dataset can be assessed in (Moura et al., 2020).

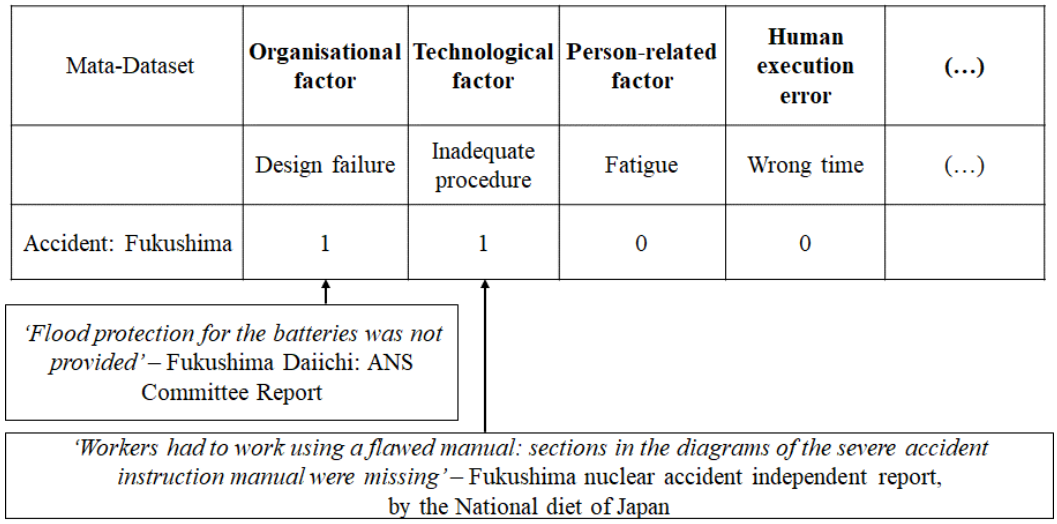


Figure 1. Human categorization analysis of accident reports issued for Fukushima nuclear accident (Daiichi, 2012; Fukushima Nuclear Accident Independent Investigation, 2012).

### *2.3. Automated text analysis algorithms*

#### *2.3.1. Extracting and representing text features*

Before classifying a document, the text features need to be extracted to generate a *representation* of the document, capturing the properties that are important for further classification (Goldberg, 2017). There are many feature extraction methods available, but the methods that can be used to extract features from text data are mainly bag-of-words (BoW), TF-IDF and word2vec (Waykole and Thakare, 2018).

A bag-of-words model extracts features from the text, specifically the vocabulary of known words and their frequency of occurrence. The reason the model is called a 'bag' of words is that it does not consider any information about the order or structure of words. To use it on a set of documents, data is collected from text files and organised into a list, forming a vocabulary. To improve results and save computational time and memory the model ignores case, punctuation, and other frequent words that do not contain relevant information, such as stop words (e.g., 'a', 'the', 'of'). To score the known words in each file (i.e. document), their presence is marked as Boolean values (0 and 1) – thus, using the list of words previously prepared, each new file is analysed and converted into a binary vector. To extract features from files, the order of words is discarded (Brownlee, 2017a). *Bag-of-bigrams* is a special case of feature combinations that counts consecutive word sequences of a given length, which proves to be more powerful than bag-of-words, as word-bigrams are more informative than individual words. However, it is difficult to know a-priori which bigrams will be useful for a specific task, thus the modeller should assign the less important combinations previously with low weights. Bag of trigrams are also common, differently from 4-grams and 5-grams that are sometimes used for letters, but rarely for words due to sparsity issues (Goldberg, 2017).

TF-IDF (Term Frequency – Inverse Document Frequency) accounts for the frequency of each word in a set of documents and its useful to give higher scores to domain specific words, something that is considered a drawback for bag-of-words (as domain specific words which does not have higher frequency within a document may be ignored). TF-IDF reduces the score of frequent words in a document that are also frequent among all the documents, highlighting the words that are unique (Hughes et al., 2017; Waykole and Thakare, 2018).

Word2vec assumes that words that occur in the same contexts tend to have similar meanings (Goldberg, 2017), thus models constructed by word2vec algorithms will place words with common contexts next to each other in a vector space (Heidarysafa et al., 2018; Waykole and Thakare, 2018). Word2vec models are two-layer neural networks, and depending on their architecture they are able to consider nearby context words more heavily than words with distant context (i.e. continuous skip gram), or to not account for context at all (i.e. continuous bag-of-words) (Waykole and Thakare, 2018).

### *2.3.2. Classifying text features*

After the text relevant features are captured from the document and represented in a model, they are ready to be classified by a machine-learning technique. The most known and broadly tested techniques for automated text classification are the dictionary method, Naïve Bayes, support vector machines (SVM), latent Dirichlet allocation (LDA), latent semantic analysis (SMA), structural topic model (STM) (Kim et al., 2020). Aside from the dictionary method, they can be mostly divided into supervised and unsupervised learning methods (some authors further distinguish semi-supervised approaches, in which the training set contains a small amount of data with known categories and a large amount of data with unknown categories (Ratsaby and Venkatesh)). The method selection might be based on how texts are going to be classified, and if some documents have been previously classified by humans (allowing their use as examples to train the machine) (Goldberg, 2017; Kim et al., 2020). Figure 2 shows the main techniques for cases where the classification category is known and pre-defined, whereas Figure 3 shows techniques which classification category is unknown.

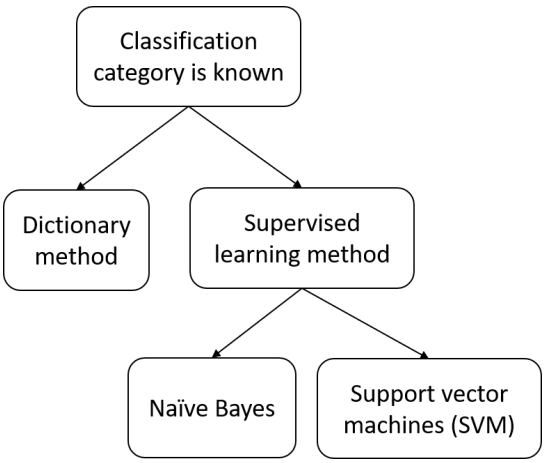


Figure 2. Most common automated text analysis techniques available when classification is known.

In dictionary-based methods, the machine uses predefined set of words to infer particular features of a text, relying on the user defined dictionary. In such methods, the categories of interest are represented by single words, which are searched by an algorithm through large bodies of text (Iliev et al., 2015; Kim et al., 2020). In the classification of organisational factors in accidents, it would be equivalent to define into the algorithm that every time the words or the expressions *work shift*, *jetlag*, *lack of sleep*, *circadian rhythm,* are identified in the text, the algorithm classifies the feature as the organisational factor of *irregular working hours*.

Naïve Bayes and support vector machines (SVM) are popular supervised learning methods for text classification. Naïve Bayes is a simple Bayesian classifier which assumes that all attributes are independent of each other, thus independent of the word context and position in the document (McCallum and Nigam, 1998; Žubrinić et al., 2013). Naïve Bayes classifiers is reported to have better resilience to missing data than SVM classifiers (Shi and Liu, 2011), what potentially makes Naïve Bayes better to analyse fragments of texts (e.g. few paragraphs) and SVM to classify whole documents (Goh and Ubeynarayana, 2017; Wang and Manning, 2012).

Support Vector Machine (SVM) is one of the most popular supervised machine-learning algorithms, due to its little need for adjustments (Matlab, 2019), and due to their excellent prediction and generalization capabilities (Arrieta et al., 2020; Goh and Ubeynarayana, 2017). They can be used for classification, regression, or other tasks such as outlier detection (Arrieta et al., 2020). The SVM algorithm constructs a hyper-plane (or a set of them) in a high-dimensional space, so that a good separation between classes is achieved by the hyperplane, that has the largest distance to the nearest training data point of any class (Arrieta et al., 2020). The simplest case, when data have only two classes, a SVM classifies data by finding the *maximum-margin hyperplane* which separates the data points of one class from those of the second class (Matlab, 2019).

The support vectors cross the data points that are closest to the hyperplane that separate the classes. As SVM is a supervised learning model, it has to be trained before it cross-validates the classifier. Only then, the trained machine can be used to predict or classify new data. SVM is usually suggested if features' interaction might be important for classification, similar to a semantic space, as learned hyperplane separates documents belonging to different topics in the input space (Žubrinić et al., 2013). Although it is usually suggested in literature that for more complex problems, other SVM kernel functions can be used to obtain more satisfactory predictive accuracy (Matlab, 2019), previous studies show that the classification performance is not always better when non-linear polynomial kernel is applied, e.g. linear kernel outperforms non-linear when applied for multi-word classification (i.e. when the context information of individual words is captured) (Zhang et al., 2008).

When the classification category is unknown, a situation represented in Figure 3, unsupervised learning methods are usually chosen to infer latent categories.

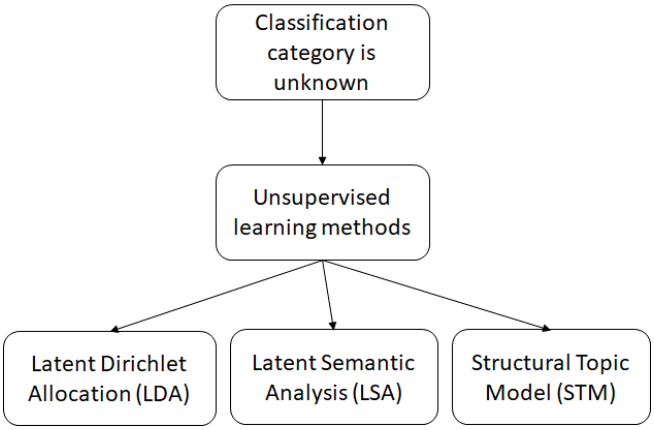


Figure 3. Most common automated text analysis techniques when classification is unknown.

The latent semantic analysis (LSA) (Robinson et al., 2015) is a quantitative text-data analysis which employs singular value decomposition, which was a precursor of the latent Dirichlet allocation (LDA) (Blei et al., 2003), the first widely used topic model (Kim et al., 2020). LSA and LDA have similar methodologies, but LSA does not depend on rigorous statistical modelling. Statistical model estimates the categories or topics based on the pattern of word co-occurrences in the text. However, although unknown, the number of classes needs to be estimated before the analysis. Structural topic model (STM) (Roberts et al., 2016) is built upon LDA (Kim et al., 2020), thus both are topic models used to discover latent themes (i.e. thematic structures in documents), being able to reveal topic proportions in each document. STM has been designed to compensate LDA weaknesses, such as possibility of incorporating metadata (e.g. investigators' nationality and year a report was issued), and modelling direct correlations among topics (instead considering them independent) (Kim et al., 2020).

#### *2.3.3. Measuring the performance of automatic text classification*

The performance of a classifier is based on its capability to correctly assign new data to the correct class. This is often represented by the true and false positives, and true and false negatives. For a binary classifier, 1 is used to represent an observed variable in a dataset while 0 represents a non-observed variable. Therefore:

- true positives occur when the true value is 1 and the model correctly predicts 1;
- false negatives occur if the true value is 1 but the model wrongly predicts 0;
- true negatives occur when true value is 0 and the model correctly predicts 0;
- and false positives occur when true value should be 0 but the model predicts 1.

The selection of the best performance metrics to observe will vary according to how false positives and false negatives predictions will cost to the study's objective. For example, the cost of false positive is higher if one is modelling how to identify spam emails (as someone can lose important information if an email is wrongly

classified as spam). However, if the intention is to model the spread of a contagious disease, the cost of having a false negative is higher (as it is more impacting to public health if a person with a disease, an actual positive, does a test which wrongly classifies them as healthy, a false negative) (Ping Shun, 2018). A confusion matrix is used to depict the four possible outcomes by comparing the true classes expected by the classes predicted (Google, 2018). On the confusion matrix plot depicted in Table 1 the rows correspond to the true class (also known as target Class), and the columns correspond to the predicted class (also known as output Class). The diagonal cells (in green) correspond to observations that are correctly classified, and the off-diagonal cells (in red) correspond to incorrectly classified observations. Some confusion matrices also show the percentage of the total number of observations in each cell, with additional columns and rows showing *accuracy*, *prediction* and *recall* measures (Matlab and Mathworks, 2018). In the example provided in Table 1 the confusion matrix indicates only the observations: 6 true positives, 2 false negatives, 1 false positive and 30 true negatives. Confusion matrices are even more useful if many variables are being classified, as it provides handy information on which classes are mostly misclassified to what other classes (Heidarysafa et al., 2018).

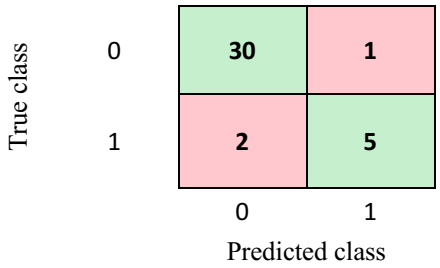


Table 1. Confusion matrix example.

There are four main metrics to evaluate model performance according to true and false predictions: *accuracy*, *precision*, *recall*, and *F-measures* score (Goh and Ubeynarayana, 2017). *Accuracy* is the fraction of correctly predicted data points out of all predictions and defined as follows:

$$Accuracy = \frac{(\text{true positives} + \text{true negatives})}{(\text{true positives} + \text{true negatives} + \text{false positives} + \text{false negatives})} \quad (1)$$

The potential problem of relying solely in accuracy is that it can be largely contributed by a large number of true negatives [44], such as when dealing with imbalanced data (a dataset which has many more instances of certain classes than others) (Sun et al., 2009).

*Precision* is a good measure to indicate the proportion of positive identifications that are actually correct, or to monitor when the cost of a false positive is high (Google, 2018; Ping Shun, 2018). *Precision* is equal to 1.0 if the model produces no false positives and defined as follows:

$$Precision = \frac{(\text{true positives})}{(\text{true positives} + \text{false positives})} \quad (2)$$

When the cost of false negative is high, the *Recall* metric is a good measure to indicate if the proportion of actual positives are identified correctly (Google, 2018; Ping Shun, 2018). A model that produces no false negatives has a *recall* of 1.0. The recall metric is defined as:

$$Recall = \frac{(\text{true positives})}{(\text{true positives} + \text{false negatives})} \quad (3)$$

*F-measures* are useful if a balance between *precision* and *recall* is needed (Ping Shun, 2018), as empirical studies of retrieval performance have shown a tendency for *precision* to decline as *recall* increases [47]. It is also a good measure if the true classes present an uneven distribution such as a large number of true negatives (Ping Shun, 2018). If false negatives and false positives are equally costly, $F_1$ score represents the harmonic mean between recall and precision:

$$F_1 = 2 \cdot \frac{(\text{Precision} \cdot \text{Recall})}{(\text{Precision} + \text{Recall})} \quad (4)$$

However, if false negatives and false positives are not equally costly, $F_\beta$ measure might be indicated as it is an abstraction of the *F-measure* where the balance of *precision* and *recall* are controlled by a coefficient called β. If false negatives cost more, β > 1; if false positives are more costly, β < 1 (He and Ma, 2013).

$$F_\beta = (1 + \beta^2) \cdot \frac{(\text{Precision} \cdot \text{Recall})}{(\beta^2 \cdot \text{Precision} + \text{Recall})} \quad (5)$$

Using the example given in the confusion matrix in Table 1, the *accuracy* of the model would be 92%, precision would be 86%, recall would be 75%, and using the results of *precision* and *recall* the $F_1$ score would be 80%.

Performance metrics may present different results depending on the size and on the randomised sample used for training and testing sets. To minimise the randomised sample effect many studies present the metrics by variable (or sets of variables) instead by overall indicators (Goh and Ubeynarayana, 2017; Grech et al., 2002; Heidarysafa et al., 2018; Zhang et al., 2019). The difference in performance metrics can be more transparently depicted by error estimates (Ribeiro et al., 2020). The need for smaller uncertainties between estimates can also define the size of training and testing sets. Some machine-learning practitioners even suggest to have larger testing sets than what is normally recommended, in order to increase the confidence in model predictions (not only because the error estimates of performance metrics decrease, but because the user can actually see how the model works for more samples) (Malato, 2015).

## 3. Methodology

Support vector machine was proposed to automatically read and classify accident reports into potential human factors, with the support of Bag-of-Words model for data collection. The model was trained and tested using data from MATA-D. This section better describes the dataset used and the procedures applied to train and test the models.

### *3.1. Dataset*

The classification tool was trained using the data from Mata-D. The decision was based on the conceptual advantaged of potential cross-learning lessons from accidents in different sectors, but also brought two technical advantages regarding machine-learning techniques. Firstly, the majority of accident reports were available to train and test the machine against the opinion classified by experts. Secondly, the dataset had a specific taxonomy, which simplified the decision on the automated text technique to choose.

The type of documents analysed were accident investigation reports, all in English, with an average size of two hundred pages. The accidents described in those reports had happened in different industry sectors with similar complexity regarding the interaction within humans, technology, and organization, such as: aviation, chemicals factory, construction, food, oil & gas (exploration, refinery, petrochemical), metallurgical, nuclear, terminals and distribution and waste treatment plant. The documents chosen were the same used to construct a dataset of 238 reports classified into a human reliability taxonomy as described in (Moura et al., 2016), known as MATA-D which can be assessed in (Moura et al., 2020).

Table 2 shows the taxonomy used, the classification scheme developed for a human reliability method known as CREAM (cognitive reliability and error analysis method) (Hollnagel, 1998). This taxonomy comprises human errors and performance shaping factors (PSFs) such as organisational, technological, and individual factors. CREAM's taxonomy has the benefit of serving both accident analysis and risk analysis purposes. Thus, by continuously updating the dataset with new accident investigation reports, the dataset will provide risk and reliability analysis with better predictions of which combinations of factors mostly trigger accidents. Although MATA-Dataset contained information on how 238 accident reports had been labelled against CREAM taxonomy, only the publicly available reports were used to train and test the virtual classifier in the present study: a total of 106 reports.

| Organisational Factors | Technological Factors | Individual factors | Human Execution Errors |
|---|---|---|---|
| Communication failure | Equipment failure | **Permanent related** | Wrong time |
| Missing information | Software fault | Functional impairment | Wrong type |
| Maintenance failure | Inadequate procedure | Cognitive style | Wrong Object |
| Inadequate quality control | Access limitations | Cognitive bias | Wrong place |
| Management problem | Ambiguous information | | |
| Design failure | Incomplete information | **Temporary related** | **Cognitive function failures** |
| Inadequate task allocation | Access problems | Memory failure | Observation missed |
| Social pressure | Mislabelling | Fear | False Observation |
| Insufficient skills | | Distraction | Wrong Identification |
| Insufficient knowledge | | Fatigue | Faulty diagnosis |
| Temperature | | Performance Variability | Wrong reasoning |
| Sound | | Inattention | Decision error |
| Humidity | | Physiological stress | Delayed interpretation |
| Illumination | | Psychological stress | Incorrect prediction |
| Other | | | Inadequate plan |
| Adverse ambient conditions | | | Priority error |
| Excessive demand | | | |
| Inadequate workplace layout | | | |
| Inadequate team support | | | |
| Irregular working hours | | | |

Table 2. Taxonomy of human factors adopted in MATA-Dataset based on CREAM classification scheme.

As the reports in the MATA-Dataset addressed different industry sectors, they presented different formats and vocabularies. The format changed not only in terms of number of pages, but also in terms of reproduceable sections in a corpus. The vocabularies varied not only on specificity of the different industrial sectors, but also in terms of taxonomy applied usually connected to the investigation methodology. This research used three different datasets: the first contained 106 publicly available reports (public at the time of the research), the second was a subset of the first dataset with 57 CSB reports (U.S. Chemical Safety and Hazard investigation board), and the third was another subset of the first dataset with 20 reports issued by NTSB (U.S. National Transport Safety Board). CSB is an U.S. independent government agency that investigates mainly industrial chemical accidents, covering accidents not only in chemical factories, but also in its branches (e.g., oil & gas, food, and metallurgical industries). NTSB is also an independent U.S. government agency, which investigates accidents in transportation, such as aviation, and including terminals and distribution. CSB and NTSB were chosen due to their larger number of reports in MATA-Dataset and due to their systematically organised and repetitive format (e.g., similar chapters titles and same order of chapters), which is potentially positive considering the training of a supervised learning technique.

The three datasets generated three different models: *all reports*, *CSB* and *NTSB* models. The reports were randomly split into a training-testing ratio of 80-20%, therefore generated a training set of 85 reports and a testing set of 21 reports for *all reports* model, 46 to train and 11 to test reports in CSB model, and 16 to train and 4 to test reports in NTSB model. The decision of choosing between an 80-20% split instead of a 90-10% was taken to increase the confidence in the results as suggested in (Malato, 2015).

### *3.2. Machine-learning technique*

As the classification of the category is known (i.e., predefined taxonomy), and the dataset was previously labelled by experts, a supervised learning method is the most adequate, short-listing the decision to Naïve Bayes or Support Vector Machine. It has been proven that Naïve Bayes classifiers perform better with missing data (Shi and Liu, 2011), and therefore it might be a good choice to identify human factors interactions in major accidents that are considered rare and uncertain events (Morais et al., 2020). However, SVM has the potentiality to better capture features interactions (Žubrinić et al., 2013) and better classify larger documents (Wang and Manning, 2012). Therefore, as interaction patterns has been observed between MATA-D factors in (Moura et

al., 2017b) and the aim is to apply the tool to accident reports with 200 pages on average, an SVM model with a linear kernel has been chosen for classification. Bag-of-words was selected as the feature extraction tool to pre-process the features to be classified by SVM. The choice was not only due to its recognised simplicity and flexibility (Waykole and Thakare, 2018), but also because the intention to classify accident reports with no specific sector or domain suggested that it was better not to use models that capture too much the context from the training set into account – to avoid giving much higher importance to sector specific words or set of words (Goldberg, 2017).

The resulting automated text recognition and classification tool is referred to as the *human factors' virtual classifier*. A simplified workflow of the proposed approach is shown in Figure 4.

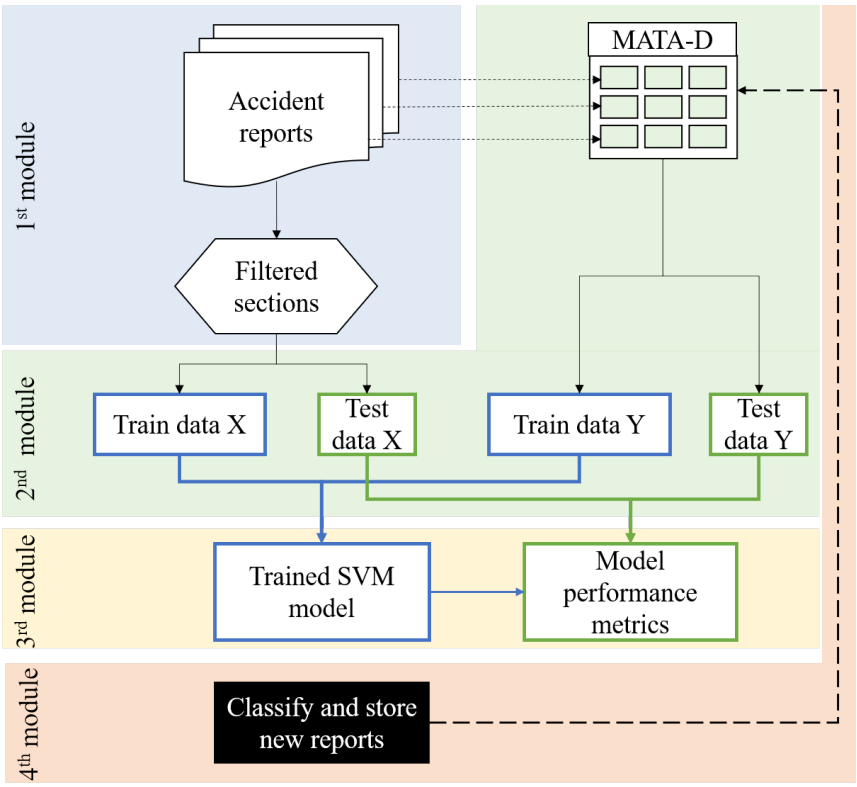


Figure 4. Simplified workflow of the human factor's virtual classifier.

In the first module, accident investigation reports were analysed. The documents in portable document format (i.e., files with PDF extension) were processed to check if the text in pdf files were recognised by the machine and, if not, an optical character recognition software (OCR) was used to convert them to text files – an important step for relatively old accident reports. After this pre-treatment, the tool scanned the accident reports, and their texts were sent to the next module. In the implemented version, the semi-supervised approach gave the users the option to manually identify relevant sections, which was the option used in this study. Otherwise, the most likely start and end of the targeted sections, *recommendation* and *lessons learned,* would be identified by a confidence scoring system (a basic algorithm, tailored for this project, which defines a dictionary of the most likely start and end target words in major accident reports), and these sections would be the output to the next module. Finally, the text was pre-processed to clean punctuation, stop words, and reduce words to their stem (e.g., 'testing' was reduced to 'test').

In the second module, using another confidence scoring system, the tool took each accident report's file name and found the most likely corresponding entry in the MATA-D. For this reason, the accident reports had equally assigned names in dataset and correspondent PDF file. This gave the machine-learning component the desired output for each accident report, which was a combination of selected section texts and their known human factors. Then, the selected text was converted into bag-of-words objects (X in Figure 4, forming the

input of the model), and the factors extracted from the MATA-D (Y in Figure 4, served as the output of the model). The module partitioned the data into a training set (80% of total) and a testing set (20% of total).

In the third module, the model based on SVM was trained and tested using data input from the previous two modules. Finally, the parameters of the classifier were recorded and overall performance metrics (i.e., *accuracy*, *precision*, *recall* and $F_1$ score) were calculated based on test sets in all categories, as well as a confusion plot generated. Only then, the tool was prepared to be used in the next module.

The fourth and final module of the tool allowed users to add a new report that was not yet part of the MATA-D. The result was a list of the human reliability factors identified by the tool (an array of the predicted positive factors), a small table with all positives and negatives predictions (the 53 factors of the chosen taxonomy), and a word cloud of the most relevant words in the report.

### *3.3. Implementation*

All the computational work was carried out using MATLAB software, and supported by the *text analytics toolbox*, which used the bag-of-words model to extract text strings from files and prepare data for the machine-learning algorithm. The *MATLAB statistics and the machine-learning toolbox* was used to transform text inputs into binary classification adopting the Support Vector Machine. Data was extracted from the Excel based MATA-Dataset, while the accident report were in portable document format (i.e., PDF extension). The text recognition software embedded in Adobe Acrobat Pro was used to convert text-images to text-strings in cases where original reports had been saved as images (e.g. relatively old accident reports, such as the Public Inquiry into the Piper Alpha Disaster [51]). Computational times to evaluate a new report, including the machine training time, took around 63 seconds (using all reports), 28 seconds (using CSB reports), and 19 seconds (using NTSB reports), using a laptop configured with Intel® Core™ i5-8265U CPU @ 1.60GHz and 16.0 GB of RAM.

The classification tool was implemented on a user-friendly web interface known as *Virtual Raphael* (after the name of the expert that had conceptualized and co-created MATA-D), where the reader can classify their own accident report online, without the need to save it to the database. Together with the results a message is displayed to remind that the human factors outputs are just an indication to support the user, and that they potentially present a similar *accuracy*, *precision*, *recall* and $F_1$ score of the test set shown in this study.

The classifier tool is freely available at the following web address: https://cossan.co.uk/private/incident_classification/. The web-interface, coded in JavaScript, links three main components: the MATA-D dataset, the public accident reports, and a collection of six Matlab scripts. The dataset and all the codes used in this work are also available to those readers and researchers that want to replicate the experiment or to do their own improvements:

- The dataset MATA-D with labelled classifications of each report is available at University of Liverpool's data repository, available at: https://doi.org/10.17638/datacat.liverpool.ac.uk/1018 [12].
- The links and references to the public accident reports classified in the MATA-D and used for the training and testing sets by the Virtual Raphael classifier are available from the Cossan website. However, due to property issues, they are not shared in their pdf formats
  The source code of the methods is available from the GitHub repository of the Cossan software: https://github.com/cossan-working-group/VirtualRaphael/.

### *3.4. Performance*

To measure the performance of the Virtual Human Factors classifier, the binary classifications available in MATA-D were used as target classes. Four performance metrics were selected: *accuracy*, *precision*, *recall* and $F_1$ score. The selection took into consideration that a typical accident in MATA-D is largely contributed by a large number of true negatives (an average of 46 negatives out of 53 categories were identified among all the reports), which might be classified as an imbalanced dataset. In those cases, $F_1$ score is considered a better metric than *accuracy* (if recall and precision are considered equally important).

The metrics were used to evaluate and compare the three trained classifiers using all reports, using the CSB reports and using the NTSB reports, respectively. To calculate them, ten randomly selected reports from the database were taken, maintaining constant the size of the samples and the training-test split. For each random sample generated, the training and testing sets were the same for the 53 category models created. The confusion matrices used to compare the true classes from MATA-D with the predicted classes are presented in Table 3 (all reports model), Table 4 (CSB reports) and Table 5 (NTSB reports). The green numbers represent the true positives and true negatives, while the red numbers are the false positives and false negatives – considering the cumulative sum of predicted results from 10 random training sets. The values in the tables indicate the counting of positive and negative classifications for all the reports.

| True class | | Predicted class: 0 | Predicted class: 1 |
|---|---|---|---|
| | 0 | 8720 | 565 |
| | 1 | 994 | 851 |

Table 3. Confusion matrix of all reports' model predictions (cumulative sum of ten different samples).

| True class | | Predicted class: 0 | Predicted class: 1 |
|---|---|---|---|
| | 0 | 4730 | 198 |
| | 1 | 458 | 444 |

Table 4. Confusion matrix of CSB reports' model predictions (cumulative sum of ten different samples).

| True class | | Predicted class: 0 | Predicted class: 1 |
|---|---|---|---|
| | 0 | 1608 | 156 |
| | 1 | 194 | 162 |

Table 5. Confusion matrix of NTSB reports' model predictions (sum of ten different samples).

The performance metrics were calculated using Eqs (1)-(4) and summarised in Table 6. The classifier model trained and tested with CSB reports obtained the best performance in all four metrics.

| | **all reports** | **CSB reports** | **NTSB reports** |
|---|---|---|---|
| **Accuracy** | 86% | 89% | 83% |
| **Precision** | 60% | 69% | 51% |
| **Recall** | 46% | 49% | 46% |
| **F1 score** | 52% | 58% | 48% |

Table 6. Performance metrics according to confusion matrices cumulative sum of 10 randomly selected report from the database.

Instead of measuring the performance based in the predictions' cumulative sums, it was also useful to analyse how the performance metrics had varied according to different training and testing sets. Therefore, Table 7 shows the minimum and maximum results achieved by the performance metrics, as well as their mean and standard deviation (SD) if the ten random samples were considered separately. It was possible to observe that the results in Table 6 matched almost completely with the performance metrics mean values in Table 7.

| | **All reports** | | | | **CSB reports** | | | | **NTSB reports** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Min | max | mean | SD | min | max | mean | SD | Min | max | mean | SD |
| **Accuracy** | 83% | 89% | **86%** | 2% | 87% | 91% | **89%** | 1% | 80% | 88% | **83%** | 3% |
| **Precision** | 51% | 69% | **60%** | 6% | 64% | 77% | **69%** | 4% | 37% | 67% | **51%** | 9% |
| **Recall** | 40% | 53% | **46%** | 4% | 42% | 55% | **49%** | 5% | 36% | 72% | **47%** | 10% |
| **$F_1$ score** | 45% | 60% | **52%** | 4% | 52% | 61% | **57%** | 3% | 38% | 57% | **48%** | 5% |

Table 7. Performance metrics of ten randomly selected report from the database considered separately.

In this study, the linear SVM model trained with all public reports achieved a mean of 86% in the *accuracy,* 60% for the *precision*, 46% in the *recall* and 52% using the $F_1$ score. Table *7* shows a slightly higher performance when the model was trained using only the CSB reports, which might be explained by their similarity of format and industry sectors. The results obtained had performed similarly to the benchmarked studies, as shown in the discussion section of this paper.

Another important type of performance is the training time required by the machine-learning algorithm. The elapsed time taken for the linear SVM to train and test with all reports was 63 seconds, with CSB reports was 28 seconds, and 20 seconds with the NTSB reports– all using the laptop configuration described in the methodology section.

Word clouds were used in this research on an attempt to inspect the bag-of-words contents from the training and testing sets in the different models, in order to better understand their performance. Figure 5, Figure 6, and Figure 7 provide visualisation to the more frequent words in training and testing sets bag-of-words for all reports, CSB reports and NTSB reports.


Figure 5. Word cloud for the trained model all reports.


Figure 6. Word cloud for the CSB model.


Figure 7. Word cloud for the NTSB model.

## 4. Case studies

In order to test the model in new accident reports (i.e., not yet on Mata-D), two investigation reports from different industry sectors (aviation and oil & gas) were chosen to be analysed and classified by the same expert that originated the dataset. The results of the automated classification were not shown to him before the task, to avoid him to get biased. Many tests were conducted prompting the automated tool to analyse different sections of each report, to see if the analysis of different chapters impacted the results in different ways. The results shown in Table 8 and Table 11 present the results when the tool analysed the full report.

### *4.1. Aviation case study – 2018 Boeing 737 MAX 8 AIRCRAFT final accident report*

On October 2018, an accident with a Lion Airline aircraft, led to 189 fatalities (KNKT, 2019). Five months later, in 2019, an Ethiopian Airlines plane crashed minutes after take-off, killing all 157 onboard (Marks and Dahir, 2020). The fact that both accidents involved the same aircraft model, a Boeing 737-8 MAX, had concerned civil society and safety regulators about the possible common flaws, which resulted in all 387 planes with same model grounded globally (BBC, 2019). The two events have been famously known by the potential design flaws of the Manoeuvring Characteristics Augmentation System (MCAS) which might have mislead the pilots' actions (Chronopoulos and Guzman).

Differently from the first test of the tool performed on the preliminary accident report (Morais et al., 2019), this research tested the machine-learning tool on the final accident report of the Lion Air Aircraft flight, issued on October 2019 (one year after the accident) (KNKT, 2019). Although the final accident report of Ethiopian airlines was reportedly issued (Marks and Dahir, 2020), the link was not accessible for unknown reasons until the date this paper was submitted to reviewers, thus not included in this research (Google, 2018; Zhang et al., 2019). For the classification of the Lion Airline report, the three different training sets were also pursued (all publicly available reports, all CSB reports, and all NTSB reports). The final accident report was previously classified by the same experts which have classified MATA-Dataset within the CREAM human factors taxonomy, in order to compare their similarity in new reports. Table 8 shows the comparison between human factors classifications obtained with human coding and different training sets. The complete report was considered (from 'SYNOPSIS' to '6 APPENDICES').

The table was colour coded according to the legend below to help the reader understand how the model prediction metrics were calculated. It also helps to show what predictions the authors considered more important for this study (the darker the colour, the more important).

( ) True positives: dark green (expert classified as '1' and machine predicted correctly as '1')
( ) True negatives: light green (expert classified as '0' and machine predicted correctly as '0')
( ) False negatives: dark red (expert classified as '1', but machine wrongly predicted as '0')
( ) False positives: red (expert classified as '0', but machine wrongly predicted as '1')

| | | | | Expert | all reports | CSB reports | NTSB reports |
|---|---|---|---|---|---|---|---|
| HUMAN | Action | Execution (Error Modes) | Wrong Time | 1 | 0 | 0 | 0 |
| | | | Wrong Type | 0 | 0 | 0 | 0 |
| | | | Wrong Object | 0 | 0 | 0 | 0 |
| | | | Wrong Place | 1 | 1 | 0 | 1 |
| | Specific Cognitive Functions | Observation | Observation Missed | 0 | 0 | 0 | 0 |
| | | | False Observation | 0 | 0 | 0 | 0 |
| | | | Wrong Identification | 0 | 0 | 0 | 0 |
| | | Interpretation | Faulty diagnosis | 1 | 1 | 0 | 1 |
| | | | Wrong reasoning | 0 | 0 | 0 | 0 |
| | | | Decision error | 0 | 0 | 0 | 0 |
| | | | Delayed interpretation | 1 | 0 | 0 | 0 |
| | | | Incorrect prediction | 0 | 0 | 0 | 0 |
| | | Planning | Inadequate plan | 1 | 0 | 0 | 0 |
| | | | Priority error | 1 | 0 | 0 | 0 |
| | Temporary Person Related Functions | | Memory failure | 0 | 0 | 0 | 0 |
| | | | Fear | 0 | 0 | 0 | 0 |
| | | | Distraction | 1 | 0 | 0 | 1 |
| | | | Fatigue | 0 | 0 | 0 | 0 |
| | | | Performance Variability | 0 | 0 | 0 | 0 |
| | | | Inattention | 0 | 0 | 0 | 0 |
| | | | Physiological stress | 0 | 0 | 0 | 0 |
| | | | Psychological stress | 0 | 1 | 0 | 0 |
| | Permanent Person Related Functions | | Functional impairment | 0 | 0 | 0 | 0 |
| | | | Cognitive style | 0 | 0 | 0 | 0 |
| | | | Cognitive bias | 0 | 0 | 0 | 0 |
| TECHNOLOGY | Equipment | | Equipment failure | 1 | 1 | 0 | 0 |
| | | | Software fault | 0 | 0 | 0 | 0 |
| | Procedures | | Inadequate procedure | 1 | 1 | 1 | 1 |
| | Temporary Interface | | Access limitations | 0 | 0 | 0 | 0 |
| | | | Ambiguous information | 1 | 0 | 0 | 0 |
| | | | Incomplete information | 1 | 0 | 0 | 0 |
| | Permanent Interface | | Access problems | 0 | 0 | 0 | 0 |
| | | | Mislabelling | 0 | 0 | 0 | 0 |
| ORGANISATION | Communication | | Communication failure | 1 | 0 | 0 | 0 |
| | | | Missing information | 1 | 1 | 0 | 0 |
| | Organisation | | Maintenance failure | 1 | 1 | 1 | 0 |
| | | | Inadequate quality control | 1 | 1 | 1 | 1 |
| | | | Management problem | 1 | 0 | 0 | 0 |
| | | | Design failure | 1 | 1 | 1 | 1 |
| | | | Inadequate task allocation | 1 | 1 | 1 | 1 |
| | | | Social pressure | 0 | 0 | 0 | 0 |
| | Training | | Insufficient skills | 1 | 1 | 1 | 1 |
| | | | Insufficient knowledge | 1 | 1 | 0 | 0 |
| | Ambient Conditions | | Temperature | 0 | 0 | 0 | 0 |
| | | | Sound | 0 | 0 | 0 | 0 |
| | | | Humidity | 0 | 0 | 0 | 0 |
| | | | Illumination | 0 | 0 | 0 | 0 |
| | | | Other | 0 | 0 | 0 | 0 |
| | | | Adverse ambient conditions | 0 | 0 | 0 | 0 |
| | Working Conditions | | Excessive demand | 1 | 0 | 0 | 0 |
| | | | Inadequate workplace layout | 0 | 0 | 0 | 0 |
| | | | Inadequate team support | 1 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| Irregular working hours | 0 | 0 | 0 | 0 |
| Sum of true positives | | 11 | 6 | 8 |
| Sum of true negatives | | 30 | 31 | 31 |
| Sum of false positives | | 1 | 0 | 0 |
| Sum of false negatives | | 11 | 16 | 14 |
| Accuracy | | 77% (79%) | 70% | 74% |
| Precision | | 92% (100%) | 100% | 100% |
| Recall (or true positive rate) | | 50% | 27% | 36% |
| $F_1$ Score | | 65% (67%) | 43% | 53% |

Table 8. Virtual expert trained using different report set vs. expert classification for Lion Airline accident report (Boeing 737-8MAX).

According to Table 8, the model trained with all reports retrieved the best *accuracy*, *recall* and $F_1$ score. Only the precision was slightly lower than those obtained using the CSB and NTSB reports. When the classifier is trained with all reports the following factors were observed in the Lion Air accident operating with the Boeing 737 MAX: human error of execution of wrong place (i.e. action out of sequence); the cognitive function failure of faulty diagnosis; the technological factors of equipment failure and inadequate procedure; the organisational factors of missing information, maintenance failure, inadequate quality control, design failure, inadequate task allocation, insufficient skills, insufficient knowledge. The confusion matrices for the three models are presented in Table 9.

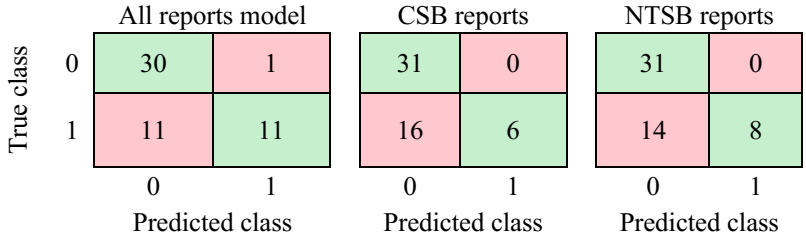


Table 9. Confusion matrices for the Boeing 737 MAX accident report predictions.

The report was also classified after selecting its potentially more important sections, which carried more information about the accident causes (the report initial information was discarded, as it contained overall info about the plane and not about the accident). For all three models, the performance metrics obtained are mostly similar to the analysis of the whole report, with slight improvement only for all reports model in terms of *accuracy* (79%), *precision* (100%) and $F_1$ score (67%). Table 10 shows the results after grouping the model outputs for all the 53 factors into 4 main groups (i.e., human errors, individual factors, technological factors, and organisational factors).

| **All reports** | **Human errors and cognitive function failures** | **Individual factors** | **Technological factors** | **Organisational factors** |
|---|---|---|---|---|
| ***Accuracy*** | 71% | 82% | 75% | 80% |
| ***Precision*** | 100% | 0% | 100% | 100% |
| ***Recall*** | 33% | 0% | 50% | 64% |
| ***$F_1$ Score*** | 50% | 0% | 67% | 78% |

Table 10. Model performance by sets of human factors for the Boeing 737 MAX report.

The word cloud was included as it might serve as an additional support for the user to check if the information in the report is being correctly extracted or if there are problems that deserve any intervention to improve the prediction performance. It might be also important to compare the word cloud obtained with the new report (Figure 8) with the word clouds of the training and testing sets (Figure 5, Figure 6, and Figure 7).

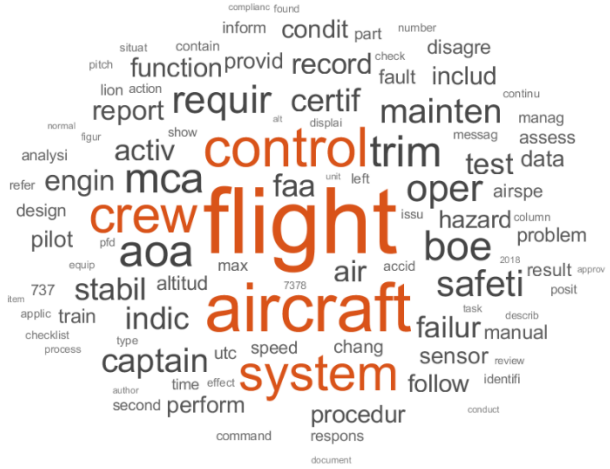


Figure 8. Word cloud for the Boeing 737 MAX accident report.

To classify the Lion Air Accident report, the algorithm took 74 seconds with the model trained with all reports, 32 seconds with model trained with CSB reports, and 30 seconds with model trained with NTSB reports (considering the training time).

### *4.2. Oil & Gas case study: FPSO Cidade de São Mateus (CDSM) accident report*

On February 2015, an explosion onboard FPSO Cidade de São Mateus killed nine, injured 26 workers, as well as caused damage to the installation, and production halt of two gas production fields up to this date (2021). The Brazilian Oil & Gas regulator (ANP) included in their investigation report root causes from the design phase to the emergency response. The FPSO (floating production, storage and offloading unit) was operated by BW Offshore in gas fields under concession to Petróleo Brasileiro S.A (Petrobras) in Brazilian waters (ANP, 2015).

The tool was also trained with the same training sets adopted to the aviation case study. The FPSO CDSM accident report was previously classified by the same experts as MATA-Dataset and Lion Airline report. Table 11 shows the comparison between human factors classifications obtained with human coding and the different training sets. The complete report was considered (from its title to 'Conclusion' chapter).

| | | | | Expert | All reports | CSB reports | NTSB reports |
|---|---|---|---|---|---|---|---|
| **HUMAN** | **Action** | Execution (Error Modes) | Wrong Time | 0 | 0 | 0 | 0 |
| | | | Wrong Type | 0 | 0 | 1 | 0 |
| | | | Wrong Object | 0 | 0 | 0 | 0 |
| | | | Wrong Place | **1** | 0 | 0 | 1 |
| | **Specific Cognitive Functions** | Observation | Observation Missed | **1** | 0 | 0 | 1 |
| | | | False Observation | 0 | 0 | 0 | 0 |
| | | | Wrong Identification | 0 | 0 | 0 | 0 |
| | | Interpretation | Faulty diagnosis | **1** | 0 | 0 | 1 |
| | | | Wrong reasoning | **1** | 0 | 1 | 0 |
| | | | Decision error | 0 | 0 | 0 | 0 |
| | | | Delayed interpretation | 0 | 0 | 0 | 0 |
| | | | Incorrect prediction | 0 | 0 | 0 | 0 |
| | | Planning | Inadequate plan | **1** | 0 | 0 | 0 |
| | | | Priority error | 0 | 0 | 0 | 0 |
| | | | Memory failure | 0 | 0 | 0 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | **Temporary Person Related Functions** | Fear | 0 | 0 | 0 | 0 |
| | | Distraction | 0 | 0 | 0 | 0 |
| | | Fatigue | 0 | 0 | 0 | 0 |
| | | Performance Variability | 0 | 0 | 0 | 0 |
| | | Inattention | 0 | 0 | 0 | 0 |
| | | Physiological stress | 0 | 0 | 0 | 0 |
| | | Psychological stress | 0 | 0 | 0 | 0 |
| | **Permanent Person Related Functions** | Functional impairment | 0 | 0 | 0 | 0 |
| | | Cognitive style | 0 | 0 | 0 | 0 |
| | | Cognitive bias | 1 | 0 | 1 | 0 |
| **TECHNOLOGY** | **Equipment** | Equipment failure | 0 | 0 | 0 | 0 |
| | | Software fault | 0 | 0 | 0 | 0 |
| | **Procedures** | Inadequate procedure | 1 | 1 | 1 | 1 |
| | **Temporary Interface** | Access limitations | 0 | 0 | 0 | 0 |
| | | Ambiguous information | 0 | 0 | 0 | 0 |
| | | Incomplete information | 1 | 0 | 0 | 1 |
| | **Permanent Interface** | Access problems | 0 | 0 | 0 | 0 |
| | | Mislabelling | 0 | 0 | 0 | 0 |
| **ORGANISATION** | **Communication** | Communication failure | 1 | 0 | 0 | 1 |
| | | Missing information | 1 | 0 | 0 | 0 |
| | **Organisation** | Maintenance failure | 1 | 1 | 1 | 0 |
| | | Inadequate quality control | 1 | 1 | 1 | 1 |
| | | Management problem | 0 | 0 | 0 | 0 |
| | | Design failure | 1 | 1 | 1 | 1 |
| | | Inadequate task allocation | 1 | 1 | 1 | 1 |
| | | Social pressure | 1 | 0 | 0 | 0 |
| | **Training** | Insufficient skills | 1 | 0 | 0 | 1 |
| | | Insufficient knowledge | 1 | 0 | 1 | 0 |
| | **Ambient Conditions** | Temperature | 0 | 0 | 0 | 0 |
| | | Sound | 0 | 0 | 0 | 0 |
| | | Humidity | 0 | 0 | 0 | 0 |
| | | Illumination | 0 | 0 | 0 | 0 |
| | | Other | 0 | 0 | 0 | 0 |
| | | Adverse ambient conditions | 0 | 0 | 0 | 0 |
| | **Working Conditions** | Excessive demand | 1 | 0 | 0 | 0 |
| | | Inadequate work place layout | 0 | 0 | 0 | 0 |
| | | Inadequate team support | 0 | 0 | 0 | 0 |
| | | Irregular working hours | 0 | 0 | 0 | 0 |
| | | Sum of true positives | | 5 | 8 | 10 |
| | | Sum of true negatives | | 35 | 34 | 35 |
| | | Sum of false positives | | 0 | 1 | 0 |
| | | Sum of false negatives | | 13 | 10 | 8 |
| | | Accuracy | | 75% | 79% | 85% |
| | | Precision | | 100% | 89% | 100% |
| | | Recall (true positive rate) | | 28% | 44% | 56% |
| | | $F_1$ Score | | 43% | 59% | 71% |

Table 11. Virtual expert vs. expert classification for FPSO Cidade de Sao Mateus accident report classification.

The model trained with NTSB reports retrieved the best *accuracy*, *precision*, *recall* and $F_1$ score. If trained with NTSB reports the following factors were observed in the oil & gas installation, the FPSO Cidade de Sao Mateus: human errors of execution of wrong place (i.e. action out of sequence); the cognitive function failures of observation missed and faulty diagnosis; the technological factors of inadequate procedure and incomplete information (related to temporary interfaces); the organisational factors of communication failure, inadequate quality control, design failure, inadequate task allocation, and insufficient skills. For another visualisation of true and false predictions, the confusion matrices for the three models are presented in Table 12.

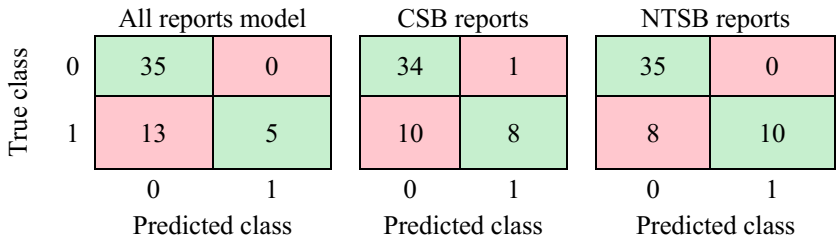


Table 12. Confusion matrices for FPSO Cidade de Sao Mateus accident report predictions.

The report was also classified after selecting its potentially more important sections, which carried more information about the accident causes (the chapters 3, 4 and 5 were discarded as they had more information on how the installation should operate rather than what went wrong). For all three models, the performance metrics obtained were mostly similar to the analysis of the whole report, with slightly improvement only for all reports model in terms of *accuracy* (77%), *recall* (33%) and $F_1$ score (50%). Table 13 shows the results after grouping the model outputs for all the 53 factors into 4 main groups (i.e., human errors, individual factors, technological factors, and organisational factors).

| **All reports** | **Human errors and cognitive function failures** | **Individual factors** | **Technological factors** | **Organisational factors** |
|---|---|---|---|---|
| ***Accuracy*** | 62% | 91% | 88% | 70% |
| ***Precision*** | 0% | 0% | 100% | 100% |
| ***Recall*** | 0% | 0% | 50% | 40% |
| ***$F_1$ Score*** | 0% | 0% | 67% | 57% |

Table 13. Model performance for FPSO Cidade de Sao Mateus by sets of human factors.

The word cloud in *Figure 9* shows that the text extracted from the full report had some frequent words that were not related to any accident cause of human factor. The words *brazilian*, *agency*, *biofuel*, and *ssm*, are related to the name of the investigation body that was repeated at the footnote in every page – thus, if possible, it would be desirable to clean footer and headers.


Figure 9. Word cloud for the full FPSO CDSM accident report.

To classify the Lion Air Accident report, the algorithm took 66 seconds with the model trained with all reports, 34 seconds with model trained with CSB reports, and 24 seconds with model trained with NTSB reports (considering the training time).

## 5. Discussion

MATA-D has the potential to incorporate the information of human reliability into risk assessments. It needs more data to increase its accuracy and reduce uncertainty. However, the data collection process of reading and classifying reports is a time consuming and challenging task, prone to errors. Therefore, this study aimed at demonstrating the capability of a machine learning tool trained using previously classified accident reports in MATA-D database to classify new accident reports with sufficient *accuracy*, *precision* and *recall*. In other words, this research investigates if machine learning is capable of accelerating the expansion of this database while maintaining the same data quality obtained with human experts. The results have shown that the automated classification of new accident reports can accelerate the data collection process, as it can reduce the time from around 3 days (when the report is classified by an expert) to around 1 minute.

### *5.1. Performance and accuracy of the automatic classifier tool*

Four performance metrics were selected to investigate the differences between expert and machine-based classification. Table 14 benchmarks the performance metrics on this study against previous studies from literature. The results are summarised in Table 14 for the classifier trained using all reports. The classifier in this study and from previous studies were trained using all the human factors and the average performance among all the factors is reported in Table 14. Additionally, only the best results available from the literature were considered. For instance, in the study of (Grech et al., 2002), when more reports were tested the precision of the method dropped from 84% to 48%, and the recall dropped from 89% to "not possible to measure".

| Metric | Test set | Aviation case study | Oil & gas case study | Previous studies |
|---|---|---|---|---|
| **Accuracy** | 86% (SD = 2%) | 77% | 75% | 44% [19]<br>75% [21]<br>90% [22] |
| **Precision** | 60% (SD = 6%) | 92% | 100% | 22% [19]<br>73% [23]<br>84% [24] |
| **Recall** | 46% (SD = 4%) | 50% | 28% | 63% [23]<br>89% [24] |
| **$F_1$ score** | 52% (SD = 4%) | 65% | 43% | 53% [22]<br>67% [23]<br>71% [21] |

Table 14. Average performance metrics for all the 53 factors versus results from literature.

The availability of an acceptable threshold for each performance metric, which could help to decide when the data collected by an automatic classification could be added to a database without corrupting its quality, is not available. The comparison in *Table 14* shows that, from the four chosen metrics, only the *recall* is below the benchmark studies.

To understand how the *recall* impacts the quality assurance of this project, it is important to understand the objectives of the classification. At a first sight the *recall* metric seems to be the best candidate for human reliability classifier, because a performance shaping factor that goes undetected prevents the allocation of resources for the risk reduction. However, a good *precision* is also important for resource allocation– for a risk assessment purpose it might be more detrimental, as resources are allocated to prevent an event that might not really contribute to the risk. In other words, both false negatives and false positives are detrimental for the

decision of partially replacing experts in the data collection. As it is not possible to achieve a *precision* and a *recall* of 100% at the same time (Buckland and Gey, 1994), it is suggested that a balance between both is achieved using the $F_1$ score. If at some part of the analysis, it is considered that the *recall* or the *precision* are not equally important, it is suggested to use $F_\beta$ with $\beta > 1$ (*recall* more important) or $\beta < 1$ (*precision* more important).

Although the test set already provided the metrics needed to benchmark the performance of the proposed automatic classifier against previous studies, the presented case studies offered additional insights into how the classifier performed. The case studies have demonstrated the applicability of the approach for different sectors (i.e., aviation and oil & gas) although the performance achieved was slightly out of the bounds established by the test set standard deviation, especially regarding the *precision* and the *recall*. Literature suggests that this difference might be decreased by using domains specific training sets (Brownlee, 2018) and this approach can be adopted to improve the *recall* for a specific industry sector. However, in this study the aim is to learn from accident occurred in different sectors and therefore training a generic classifier.

For the oil & gas case study trained with all reports, a perfect prediction (100%) has been obtained although with a low *recall* score (28%), meaning that only a few human errors and performance shaping factors were identified but no false positive.

It has also been tested whether grouping all the 53 factors into 4 main groups (i.e., human errors, individual factors, technological factors, and organisational factors) would have been able to improve the classification when the classifier is trained using all reports. For the aviation case study, as shown in Table 10, the $F_1$ score improved to 78% for organisational factors and only to 65% for technological factors (compared to the overall mean of 65% shown in *Table 14*). For the oil & gas case study, in Table 13, the F1 score of organisational and technological factors improved to 78% and 67%, respectively due to the use of an enriched training dataset with higher frequency of organisational and technological factors. For both case studies, the $F_1$ score of human errors and individual factors performed worse when analysing the factors by groups.

Surprisingly, the oil & gas case study has showed better results when the classifier was trained using only NTSB reports. Although this set contains some reports related to oil & gas terminals and distributions, the majority of reports are from the aviation sector. The expectation was that CSB reports would have provided a better training set. For the Lion Air accident report, the classifier trained with all reports performed better than those trained only with NTSB reports, which contains more aviation specific language (as can be seen by the word cloud presented in Figure 7). This result might be due to the different formats used for the reports tested, as they are from different investigation bodies.

Observing the results of the case studies, it has been noted that the majority of categories detected by the machine-learning approach were inside the 26 most significant contributing factors per cluster identified in a previous research (Moura et al., 2017b). This might suggest training the classifier using only fewer frequent categories. However, tests were performed reducing the number of categories to the 13 most frequent ones, and the results did not present significant changes, e.g., an improvement of ~5% for *precision*, *recall* and $F_1$ score, but with a deterioration of the same level in the *accuracy*. Therefore, it has been decided to keep all categories in the training set, as the main goal of this research is to expand the current MATA-D dataset using the same categories already available and therefore decrease the uncertainty associated to rare combinations of human error and performance shaping factors.

This study had not found a significant difference between the automated classifications of full reports and of reports' selected sections. Word cloud figures were provided to visualise frequent words and aided the task of inspecting which sections of the accident reports provided more relevant information.

### *5.2. Future improvements and recommendations*

One of the limitations of the current classifier is its moderate capability to identify infrequent classes. One solution is to enrich the training set with accident reports where those infrequent classes had occurred (according to an analysis provided by human factor experts) – by training the model on these classes it is expected that the overall *recall* metric will increase as more data is used. Different resampling strategies might also be used (e.g., targeting infrequent classes to resample rather than sampling the training data set randomly). Finally, algorithms that maximise the *recall* while using the *precision* metric as a constraint should also be investigated (see e.g. (Bennett et al., 2017). Solutions to strengthen learning with regards to the small class might be applied (e.g., adjusting the SVM class boundary based on kernel-alignment). Further research might also assume higher

misclassification costs applied to samples in the infrequent classes and seek to minimize high cost errors (Brownlee, 2021; Sun et al., 2009).

In addition, further development of the word cloud tool to inspect bags-of-words of each human factor category are suggested. This might also help to understand some infrequent classes. Additionally, adjustments or pre-processing on the format of accident investigation reports could potentially improve the predictions from automated classifiers. The availability of good quality accident reports will also improve the performance of automatic classifiers. For instance, accident reports should have consistent chapter enumeration, only repeated in the summary, or referred in the body text. Section titles should clearly state if the information explain the normal characteristics of the system and it should not mix important information about the accident within normal behaviour. Key information should also be provided in textual format and not only as image. Finally, the public availability of accident reports even if not in English (as translating tools are steadily getting better) would significantly contribute to the knowledge of human error.

## 6. Conclusions

A virtual human factors classifier based on machine learning has been presented to provide an automatic classification of accident reports involving human error. The approach represents an efficient way of expanding existing human reliability databases based on accident reports analysed by a machine-learning algorithm. The approach has the potential to substitute, or at least support, the classification task normally conducted by a human expert (a time-consuming process that could take weeks, depending on the complexity of the event and on the number of reports or inquiries available). The developed tool provides nearly real-time classification into a specific taxonomy able to classify a two hundred pages report in a minute (an insignificant time compared to the time required for a person to complete the same task).

The findings will be of interest for risk assessors of any industry sector that may need to learn more and faster from major accidents, as automated text analysis can help them to expand their datasets. The presented approach focused at collecting new data for the MATA-D, but the tool can easily be used with other human reliability taxonomy or to be applied to components' reliability data, as long as a labelled dataset is provided together with the text sources.

The case studies showed that the approach is robust and efficient. The performance metrics achieved are satisfactory when compared against human classification and previous studies. In addition, this is the only study which has been trained using reports from different industry sectors, and with a relatively large number of human reliability categories. The results have demonstrated the possibility of using machine-learning based approaches for helping the empirical data collection to improve human reliability analysis, and finally learning lessons from different industry sectors in an efficient and timely way.

## Acknowledgements

Caroline Morais gratefully acknowledges the Brazilian Oil & Gas regulator ANP (Agencia Nacional do Petroleo, Gas Natural e Biocombustiveis) for the support for her research. Edoardo Patelli was partially supported by the EPSRC grant EP/R020558/2 Resilience Modelling Framework for Improved Nuclear Safety (NuRes). The authors also acknowledge Mrs. Raneesha for helping to implement the virtual classifier in the Cossan website, and Jack Tully, a final year graduation student for helping to test the code.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## References

ANP, 2015. Investigation report of the explosion incident of the explosion incident occurred on 11/02/2015 in the FPSO Cidade de São Mateus Agência Nacional do Petróleo, Gás Natural e Biocombustíveis (ANP), Brazil.

Arrieta, A.B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., 2020. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion 58, 82-115.
Baybutt, P., 2016. Designing risk matrices to avoid risk ranking reversal errors. Process Safety Progress 35, 41-46.
BBC, 2019. Boeing: Which airlines use the 737 Max 8?, BBC. BBC, https://www.bbc.co.uk/news/business-47523468.
Bennett, P.N., Chickering, D.M., Meek, C., Zhu, X., 2017. Algorithms for active classifier selection: Maximizing recall with precision constraints, Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, pp. 711-719.
Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent dirichlet allocation. Journal of machine Learning research 3, 993-1022.
Brownlee, J., 2017a. A Gentle Introduction to the Bag-of-Words Model, https://machinelearningmastery.com/gentle-introduction-bag-words-model/.
Brownlee, J., 2017b. What is the Difference Between Test and Validation Datasets?, Machine Learning Mastery, https://machinelearningmastery.com/difference-test-validation-datasets/.
Brownlee, J., 2018. The Model Performance Mismatch Problem (and what to do about it), Machine Learning Mastery, https://machinelearningmastery.com/the-model-performance-mismatch-problem/.
Brownlee, J., 2021. Cost-Sensitive Learning for Imbalanced Classification, Machine Learning Mastery, https://machinelearningmastery.com/cost-sensitive-learning-for-imbalanced-classification/.
Buckland, M., Gey, F., 1994. The relationship between recall and precision. Journal of the American society for information science 45, 12-19.
CCPS, C.f.C.P.S., 2010. Guidelines for Risk Based Process Safety. John Wiley & Sons.
Chronopoulos, C., Guzman, N.H.C., Is Smartness Risky? A Framework to Evaluate Smartness in Cyber-Physical Systems, 30th European Safety and Reliability Conference and 15th Probabilistic Safety Assessment and Management Conference.
Daiichi, F., 2012. Ans committee report. A Report by The American Nuclear Society Special Committee on Fukushima.
Drupsteen, L., Groeneweg, J., Zwetsloot, G.I.J.M., 2013. Critical steps in learning from incidents: using learning potential in the process from reporting an incident to accident prevention. International Journal of Occupational Safety and Ergonomics 19, 63-77.
EASA, E., International Maintenance Review Board Policy Board, https://www.easa.europa.eu/domains/aircraft-products/international-maintenance-review-board-policy-board-IMRBPB#group-easa-downloads.
Fukushima Nuclear Accident Independent Investigation, C., 2012. The national diet of Japan. The Official Report of the Fukushima Nuclear Accident Independent Investigation Commission.
Goh, Y.M., Ubeynarayana, C.U., 2017. Construction accident narrative classification: An evaluation of text mining techniques. Accident Analysis & Prevention 108, 122-130.
Goldberg, Y., 2017. Neural network methods for natural language processing. Synthesis Lectures on Human Language Technologies 10, 1-309.
Gonçalves, F.C.C., Trabasso, L.G., 2018. Aircraft Preventive Maintenance Data Evaluation Applied in Integrated Product Development Process. Journal of Aerospace Technology and Management 10.
Google, 2018. Machine Learning Crash Course<br>, https://developers.googleblog.com/2018/03/machine-learning-crash-course.html.
Grech, M.R., Horberry, T., Smith, A., 2002. Human error in maritime operations: Analyses of accident reports using the Leximancer tool, Proceedings of the human factors and ergonomics society annual meeting, 19 ed. Sage Publications Sage CA: Los Angeles, CA, pp. 1718-1721.
He, H., Ma, Y., 2013. Imbalanced learning: foundations, algorithms, and applications.
Heidarysafa, M., Kowsari, K., Barnes, L., Brown, D., 2018. Analysis of Railway Accidents' Narratives Using Deep Learning, 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE, pp. 1446-1453.
Hollnagel, E., 1998. Cognitive reliability and error analysis method (CREAM). Elsevier.
Hughes, P., Figueres-Esteban, M., Van Gulijk, C., 2017. From negative statements to positive safety, 26th European Safety and Reliability Conference. CRC Press/Balkema, p. 307.
Iliev, R., Dehghani, M., Sagi, E., 2015. Automated text analysis in psychology: Methods, applications, and future developments. Language and Cognition 7, 265-290.
Kim, S.H., Lee, N., King, P.E., 2020. Dimensions of religion and spirituality: A longitudinal topic modeling approach. Journal for the Scientific Study of Religion 59, 62-83.
KNKT, 2019. Aircraft Accident Investigation Final Report Boeing 737-8 (MAX) Lion Mentari Airlines KNKT.18.10.35.04. KNKT, internet.
Leveson, N., 2020. Safety III: A Systems Approach to Safety and Resilience.
Lima, E.N., Benites, R.D., Mosleh, A., Martins, M.R., 2019. A Methodology to Use Multi-Objective Optimization Criteria for an Offshore Topside Production System Since the Early Design Stages, and for The Unit Life Cycle.
Malato, G., 2015. Why training set should always be smaller than test set<br>, Towards Data Science. Towards Data Science, https://towardsdatascience.com/why-training-set-should-always-be-smaller-than-test-set-61f087ed203c

Marks, S., Dahir, A.L., 2020. Ethiopian Report on 737 Max Crash Blames Boeing.
Matlab, 2019. Support Vector Machines for Binary Classification, https://www.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html.
Matlab, Mathworks, 2018. Matlab documentation for confusion chart function<br>, https://uk.mathworks.com/help/stats/confusionchart.html.
McCallum, A., Nigam, K., 1998. A comparison of event models for naive bayes text classification, AAAI-98 workshop on learning for text categorization, 1 ed. Citeseer, pp. 41-48.
Morais, C., Estrada-Lugo, D., Jacques, T., Tolo, S., Moura, R., Beer, M., Patelli, E., 2021 (in press). Robust data-driven human reliability analysis using Credal Networks (in press)<br>. Reliability Engineering & System Safety Journal.
Morais, C., Moura, R., Beer, M., Patelli, E., 2020. Analysis and estimation of human errors from major accident investigation reports. ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg 6.
Morais, C., Yung, K., Patelli, E., 2019. Machine-learning tool for human factors evaluation-application to lion air Boeing 737-8 max accident, UNCECOMP 2019 and 3rd ECCOMAS Thematic Conference. National Technical University of Athens.
Moura, R., Beer, M., Patelli, E., Lewis, J., 2017a. Learning from major accidents: Graphical representation and analysis of multi-attribute events to enhance risk communication. Safety Science 99, 58-70.
Moura, R., Beer, M., Patelli, E., Lewis, J., Knoll, F., 2016. Learning from major accidents to improve system design. Safety Science 84, 37-45.
Moura, R., Beer, M., Patelli, E., Lewis, J., Knoll, F., 2017b. Learning from accidents: interactions between human factors, technology and organisations as a central element to validate risk studies. Safety Science 99, 196-214.
Moura, R., M., B., E., P., J., L., Knoll, F., 2020. Multi-Attribute Technological Accidents Dataset (MATA-D).
OREDA, Offshore and Onshore Reliability Data, https://www.oreda.com/.
Patelli, E., Ghanem, R., Higdon, D., Owhadi, H., 2016. COSSAN: a multidisciplinary software suite for uncertainty quantification and risk management. Handbook of uncertainty quantification, 1-69.
Ping Shun, K., 2018. Accuracy, Precision, Recall or F1? Towards Data Science, https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9.
Ratsaby, J., Venkatesh, S.S., Learning from a mixture of labeled and unlabeled examples with parametric side information, Proceedings of the eighth annual conference on Computational learning theory, pp. 412-417.
Ribeiro, L.C.F., Afonso, L.C.S., Colombo, D., Guilherme, I.R., Papa, J.P., 2020. Evolving Neural Conditional Random Fields for drilling report classification. Journal of Petroleum Science and Engineering 187, 106846.
Roberts, M.E., Stewart, B.M., Airoldi, E.M., 2016. A model of text for experimentation in the social sciences. Journal of the American Statistical Association 111, 988-1003.
Robinson, S.D., Irwin, W.J., Kelly, T.K., Wu, X.O., 2015. Application of machine learning to mapping primary causal factors in self reported safety narratives. Safety Science 75, 118-129.
Sarkar, S., Maiti, J., 2020. Machine learning in occupational accident analysis: a review using science mapping approach with citation network analysis. Safety Science 131, 104900.
Shi, H., Liu, Y., 2011. Naïve Bayes vs. support vector machine: resilience to missing data, International Conference on Artificial Intelligence and Computational Intelligence. Springer, pp. 680-687.
Siegrist, J., 2011. Mixing good data with bad: how to do it and when you should not, Vulnerability, Uncertainty, and Risk: Analysis, Modeling, and Management, pp. 368-373.
Sun, Y., Wong, A.K., Kamel, M.S., 2009. Classification of imbalanced data: A review. International journal of pattern recognition and artificial intelligence 23, 687-719.
Wang, S.I., Manning, C.D., 2012. Baselines and bigrams: Simple, good sentiment and topic classification, Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 90-94.
Waykole, R.N., Thakare, A.D., 2018. A Review of feature extraction methods for text classification. IJAERD 4, 351-354.
Zhang, F., Fleyeh, H., Wang, X., Lu, M., 2019. Construction site accident analysis using text mining and natural language processing techniques. Automation in Construction 99, 238-248.
Zhang, W., Yoshida, T., Tang, X., 2008. Text classification based on multi-word with support vector machine. Knowledge-Based Systems 21, 879-886.
Žubrinić, K., Miličević, M., Zakarija, I., 2013. Comparison of Naive Bayes and SVM classifiers in categorization of concept maps. International journal of computers 7, 109-116.