Categorization Errors for Data Entry in Maintenance Work-Orders

Thurston Sexton[1], Melinda Hodkiewicz[2], and Michael P. Brundage[3]

[1,3] *National Institute of Standards and Technology, Gaithersburg, MD 20814, USA*
thurston.sexton@nist.gov
michael.brundage@nist.gov
[2] *University of Western Australia, Crawley WA 6009, AUS*
melinda.hodkiewicz@uwa.edu.au

Abstract

Manufacturing is seeing a significant push toward digitization of processes and decision making. This push is enabled by the increased availability of *data*. Yet the work of the maintenance team, one of the core subsystems in any production line, remains a largely *human* endeavor. It involves manual work on the equipment and data collection by maintainers, which itself involves free-text and pre-specified categories or controlled vocabulary, rather than collection via sensors. Often this data is un-useful, in that it does not support the digitization of work. This paper presents an approach using Human Reliability Analysis (HRA) to identify the human errors associated with entering un-useful data. A Cognitive Task Analysis (CTA) is created, based on deconstructing the individual actions and decisions performed by maintainers in this process, backed by cognitive models to ground the task analysis in theory. A breakdown of human error modes for each task is provided as a list of Unsafe Acts (UAs), along with key contextual and organizational considerations, given as performance-shaping factors (PSFs). To demonstrate usage of this CTA, initial instantiation of a common HRA framework is provided as a case study, both to estimate base human error probabilities (BHEPs) and to motivate a discussion around initial risk mitigation strategies.

1. Introduction

In Manufacturing, there is a significant push toward the digitization of processes and decision making by increasing the level of automation and networking via cyber-physical systems (the so-called "Industrial Internet of Things"), and Machine Learning methods that can parse useful patterns from the data they generate. As such, this push toward being "smart" is largely driven by the availability of this *data*; whether for analysis, decision guidance, or the training of Artificial Intelligence (AI) systems. One of the core subsystems in any production line is the maintenance team, which remains a largely *human* endeavor, despite disciplines like Prognostics and Health Management (PHM) that are making headway toward "smarter" maintenance. Consequently, the historical data needed for research and development of AI-assisted maintenance frameworks is often coming directly from humans, and is almost never in an immediately computable form, but as natural language text, full of misspellings, jargon, and abbreviations (Sexton et al., 2017).

1.1. Why Does it Matter?

The most obvious benefit to regular maintenance is reduced machine downtime, but this can come at a cost; over-scheduled maintenance costs a great deal in labor, and so a large cross-section of small-and-medium-sized enterprises (SME's) still rely on reactive maintenance (Thomas, 2018). The investigative process of the technicians is often recorded, making these records a source for data that is rich with tacit knowledge about system- and unit-level behavior. This could, in theory, be used for prognostics, diagnostics, and investigatory analysis/suggestions (Brundage et al., 2017).

The ability to quickly parse through and learn from this data, and then automatically suggest diagnostic actions, would be invaluable—especially when the time spent diagnosing is commonly longer than the time spent actually repairing the machine (Kegg, 1984). Additionally, as the experienced maintenance workforce ages and fewer young people choose to enter the field, a structured data format to preserve the "mental models" of the expert technicians is in high demand, to both assist root-cause analysis and to train new workers more efficiently (Pantförder et al., 2017; Loch & Vogel-Heuser, 2017). The unstructured data actually being produced, however, makes this type of analysis nearly impossible without highly intensive studies, and drastic behavior intervention in regards to data input and management.

1.2. Proposed Work

To deal with this, one might enforce data-entry in a graphical user-interface (GUI) into pre-specified functional categories (generally using some form of controlled vocabulary). This is not always successful, and consistent reports from industry indicate that data entry remains a process fraught with significant errors (Unsworth et al., 2011; Molina et al., 2013). These *human errors* are unintentional actions or decisions[1] (whether slips, lapses, or mistakes), which lead to undesirable outcomes —in this case, the recording of un-useful data. Here we define "un-useful data" as mistake-prone, inconsistent, or incomplete data, recorded in a way that is not sufficiently structured for use in analytics as needed by an organization. Quantifying and managing sources of human error is a primary concern of Human Reliability Analysis (HRA). Applications of HRA methods to manufacturing are ongoing (Schemeleva et al., 2012), but there remains a need for user-based, error resistant data collection within manufacturing maintenance. This paper offers a framework and methodology for applying HRA to quantify and understand human errors associated with entering un-useful maintenance work-order (MWO) data into a controlled database (DB).

2. BACKGROUND

This data-entry problem is not a new one. Many research endeavors, and many company dollars, have been spent on "solutions" to investigatory-data-entry errors to track maintenance on the shop-floor. Of course, this problem is also not limited to maintenance, or even manufacturing, and much of the current field is influenced by advances in data structuring in Medicine (though their approach has severe limitations when applied to Engineering domains, as will be discussed).

2.1. Previous work in Data Structuring

There have already been some successes in mitigating human data-entry error by automatically parsing unstructured natural language records e.g. the medical field. Patient medical records have a number of similarities with MWO's, since both outline symptoms and diagnoses, along with actions taken, all via unstructured text. Much of recent medical research has been directed toward mining text from patient records (Heinze et al., 2001; Tremblay et al., 2009; Zhou et al., 2006). However, it is important to acknowledge significant advantages when dealing in medical records, compared to the engineering domain of manufacturing maintenance: (1) data-sets are often larger, covering longer time-spans, and (2) medicine has existing controlled vocabularies with wide adoption by experts.

One approach to managing MWO data input in manufacturing comes from Computerized Maintenance Management Systems (CMMS), which have gained traction in the PHM community of late; they include names like SAP Intelligent Asset Management, and IBM's Maximo. In essence, these systems revolve around mitigating error by enforcing a strict hierarchy of known system entities and behaviors. This manifests as interacting sets of controlled vocabularies, which limit the "words" available for use in tracking work-orders.

A hallmark of this approach is the ever-present drop-down menu, where one *categorizes* from a long list of options *which thing* went wrong, and how, followed by the action-steps taken to resolve the issue. Multiple users of these systems report that such restrictions constantly lead to cluttered GUIs, compounded with a desire to select "miscellaneous" categories and explain details by hand (Sexton et al., 2017). If a "miscellaneous" classification is generally un-useful for analysis, then these systems are therefore plagued by the human errors of their own.

The other approach involves recent progress in Natural Language Processing, (e.g., Named Entity recognition, document classification, etc.) where insights can be gained by discovering latent patterns in text through Machine Learning methods. The key problem with utilizing these tools comes from the lack of *quantity*, as the number of work-orders is simply not enough for these tools to work with certainty in whatever specific manufacturing domain they occur in. The jargon and abbreviations are often specific to not only manufacturing domains, but even individual production lines or between teams of staff, making large-scale statistical inference incredibly difficult.

2.2. HRA Background

The goal of HRA methods generally is to quantify a *Human Error Probability* (HEP) for some pre-determined task. First generation methods were heavily based on reliability engineering and probabilistic risk assessment methods from Nuclear and Aviation domains, as a convergence of reliability in systems research and the burgeoning Human Factors field. However, despite some successes, a key drawback to their usage was a lack of contextual and human cognitive, behavioral, and social information. Second-generation methodologies began to appear near the turn of the century, starting with CREAM in Hollnagel, 1998. These attempt to address shortcomings of the first generation of tools, through careful structuring of their methods, usage of classification schemes for human errors, and cognitive justification for models they used in representing human action.

The process of HRA involves breaking down a "task" in question into sub-tasks that address salient levels of abstraction for analysis. These are used to apply available models of cognition at those levels, to succinctly specify the human actions taking place during the task. This is a Cognitive Task Analysis (CTA). Once this process is completed, a set of possible human errors at each sub-task, called Unsafe Acts (UAs), are

[1] distinguished from intentional *violations*

Table 1. Summary of key cognitive theory used

Decision Point	Theory	Reference
1) Relevant causal/functional relationships	Associative Strength	Fazio, Williams, & Powell, 2000
2) Organization/Categorization	Similarity-Choice	Logan, 2004

enumerated. Then, Performance Shaping Factors (PSFs) are determined, which modify UA severity per environmental, physical, or mental context during task performance. Finally, an error modelling framework for quantifying these variables is employed to calculate the actual HEP, which can be used to design risk mitigation strategies for this task.

3. THEORETICAL COGNITIVE FRAMEWORK

To begin to address these issues, it is necessary to measure the risk for human error when utilizing current systems, to inform the solutions being developed for the future. This work will focus on the specific interaction between Human agents (the technicians and operators entering data), the GUIs, and organizational context for inputting *structured* maintenance work-orders. In practice, this maybe done in multiple manners:

- Free-form (natural language processing)
- Controlled Vocabulary (DB Schema)
- Hybridized (user tagging)

Our objective is to build a guiding model of the data-entry (human+machine) system, to assist in e.g., calculating the probability of "human error" upon recording maintenance work-order cause and effect designations. Because CMMS systems are undergoing increasing adoption, and because a push for adoption of Smart Manufacturing methodologies is being felt even by small and medium enterprises (SME's), we focus on the human errors encountered when attempting to categorize a diagnosis to match a controlled-vocabulary DB schema. In this case, human error will be defined as the input of "un-useful data", i.e., data that is unstructured, and not sufficiently usable for the post hoc detection of patterns for analysis and diagnostics. As a specific example encountered quite often in our controlled-vocabulary case, this form of error is encountered when a technician selects a field that enables his ability to skip categorization (i.e., "misc", or "other"), and enter free-form text anyway (bypassing the intent of such a system).

3.1. Cognitive Modeling of Activity

Starting from the occurrence of—and subsequent expert diagnosis contained within—a maintenance work-order, this probability for error can be quantified through the modeling of human behavior under such circumstances, corresponding roughly to two "decision points": 1) Causal and/or functional *relationship* identification, through memory activation by Associative strength; and 2) Categorization and other organizational *decisions* through similarity-choice selection. These are discussed here in more detail, with an overview[2] presented in Table 1.

Category-Item Associative Strength When a technician has diagnosed a work-order's core problem, according to his own mental-model of the machine or system, the prospect of recording this information, no matter the form, requires him to conjure the related features *associated* with the breakage. This is, in some ways, the core function of keeping such maintenance work-order records: encapsulating the pattern of system behavior surrounding some given breakage. Say a technician is inputting a description and resolution for an issue like "Leaking seal at machine #32". He might subsequently bring to mind the surrounding circumstances, like "hydraulic fluid," "operator needs training," or "accumulator at low pressure." In this way, multiple concepts — broken down as objects and attributes — become activated *Items* from memory that surround the *Category* of "machine #32 seal brokenness", in much the same way that Fazio et al., 2000 treat "Auto-tires," with respect to "Goodyear" and "Firestone".

Similarity-Choice Once a sufficient number of related features have been recalled, the technician will be required to *select* the ones that maximally meet his needs. This is a key use of Similarity Choice Theory, originally set down as such by Luce, 1977, though the topic has been loosely known since at least Thurstone, 1927, and has seen many modifications and applications since (Luce, 1977). This theory posits that the probability for a human to select some feature out of several is the ratio of its importance to the sum of importance weights across the feature-space. These weights are related to the evidence that each feature "belongs" to the category of choice. It is important to note that the technician's "needs" are both categorical and temporal — he will not want to spend more then the minimum time necessary to translate his mental model (crystallized in the previous step) into this data-recording GUI. Therefore, it might often be the case with more ambiguous category-relations that error (i.e., free-form text input) will be *inviting* when under severe time-constraints.

[2] The specific, cited sources in table 1 are only a selection, meant to be well-representative of a larger body of work on the subject

3.2. Theory Interaction and Implementation

There exists a clear directionality in the use of theories 1 and 2, starting with the recall of features related to a work-order from the technician's mental-model, followed a determination of specific features and their mapping to some data management system. The process culminates in a MWO record. This "pipeline" may be followed by any of the three data-input schemes discussed in previously.

While our study focuses on a mapping to controlled vocabulary sets, there is some indication of decreased cognitive load by recording natural-language/free-form input, which could be due to step #2 (categorization) being unnecessary (and therefore, "skipped"). Rasmussen, 1983 mentions that natural language is incredibly flexible in its ability to cover a wide range of abstraction levels, while sacrificing categorical/contextual information (Rasmussen, 1983). This makes natural-language input an almost direct transcription from the category-item associations step to the actual act of recording in an iterface/GUI. Controlled-vocabulary systems, on the other hand, necessitate an act of classification—of *translation*—so that the technician's mental model will fit maximally into one of several pre-defined contexts, or (to use Rassmussen's terminology), "symbols".

The internal item-category relationship must now become an arbitrator over a feedback loop between searching the GUI for information, and the act of classification itself. The search for input mechanisms in a GUI will have bidirectional influence on the the categorization process: one direction if, for example, chosen features are not found in a drop-down list; or the other, if a salient feature is actually remembered by seeing it in the GUI. This dynamic process can, in turn, modify the mode of information processing from controlled-search for categories, to automatic detection (e.g., of habitual free-form input). This categorization loop is therefore a central defining factor in categorization error, and will be discussed in greater detail as part of the cognitive task analysis outlined in Section 4.

As for the hybrid data-entry method, it is important to note that a lack of this feed-back loop dynamic is mentioned in literature as a distinct benefit to tagging-based systems, which simultaneously allow free-form input to "skip" the classification step, while providing symbolic and contextual representation through larger-scale relationships called "Folksonomies (Peters, 2009).

4. TASK ANALYSIS

To map the above cognitive modeling to specific tasks encountered during the act of categorization in data-entry, we first conduct a Procedural Task Analysis (PTA). Then, using Bloom's taxonomy for categories of cognitive skills [3] , (Adams, 2015) we we build a framework for applying our cognitive theoretical models to tasks within the the PTA. This framework is outlined as task analysis flowchart in Figure 1.

4.1. Procedural Task Analysis (PTA)

To retain a suitable scope for this work, we focus on the higher-level cognitive load of decision-making *while engaging in the act of data-entry* for a controlled-vocabulary DB schema. This precludes the need for a sensory-based analysis of GUI types, along with the accessibility of a data-entry location on the shop-floor. Additionally, sufficient time is assumed to be available for the technician to attempt entering the MWO—or at least, some amount of time has already been calculated or deemed reasonable for this task. The load of calculating that time, based on external shop-floor pressures, is outside the scope of this work.

This task is largely serial in nature, with a defined problem-solving process taking place, beginning with the recall of symptoms observed, and ending with the entry of (hopefully) schema-compliant categories of MWO data. If time still remains, a technician will first identify the parts of his knowledge that are relevant to the task at hand, letting him (and others) distinguish this MWO from other types; see Figure 1 (middle). If this is a routine or standard MWO, the recalled features will map immediately to the already-learned DB categories, and the MWO will be entered. If not, the process of organization requires a categorization loop; see Figure 1 (bottom). Here, tribal knowledge is iteratively translated into candidate DB categories, until either a match is found, or time "runs out". This serial, temporal procedure is well-suited to a PTA, as opposed to a more categorical, atemporal hierarchical task analysis (HTA) (Chandler et al., 2006).

4.2. Cognitive Theory Integration

Combining this proposed PTA with the theoretical cognitive framework of Section 3 resuts in the Cognitive Task Analysis (CTA) that we use to contextualize further reliability analysis. The two chief decision points are notably of different cognitive level, with the organization/categorization-loop requiring the creation of abstract relationships between the technician's own knowledge and an imposed (and sometimes ambiguous) controlled vocabulary—Bloom's cognitive level of *synthesis*, at the least. Meanwhile, the first decision is primarily concerned with the recall of, and associative strength between, elements of the technician's knowledge alone (*comprehension* and *analysis*). As mentioned previously, analysis in this work would be performed with preliminary cognitive theory from Sinha, 2005 in mind, which indicates that *tagging* may reduce cognitive load compared to categorization for the task

[3] The six categories defined by Bloom, in order of *increasing* complexity, are knowledge, comprehension, application, analysis, synthesis, and evaluation.

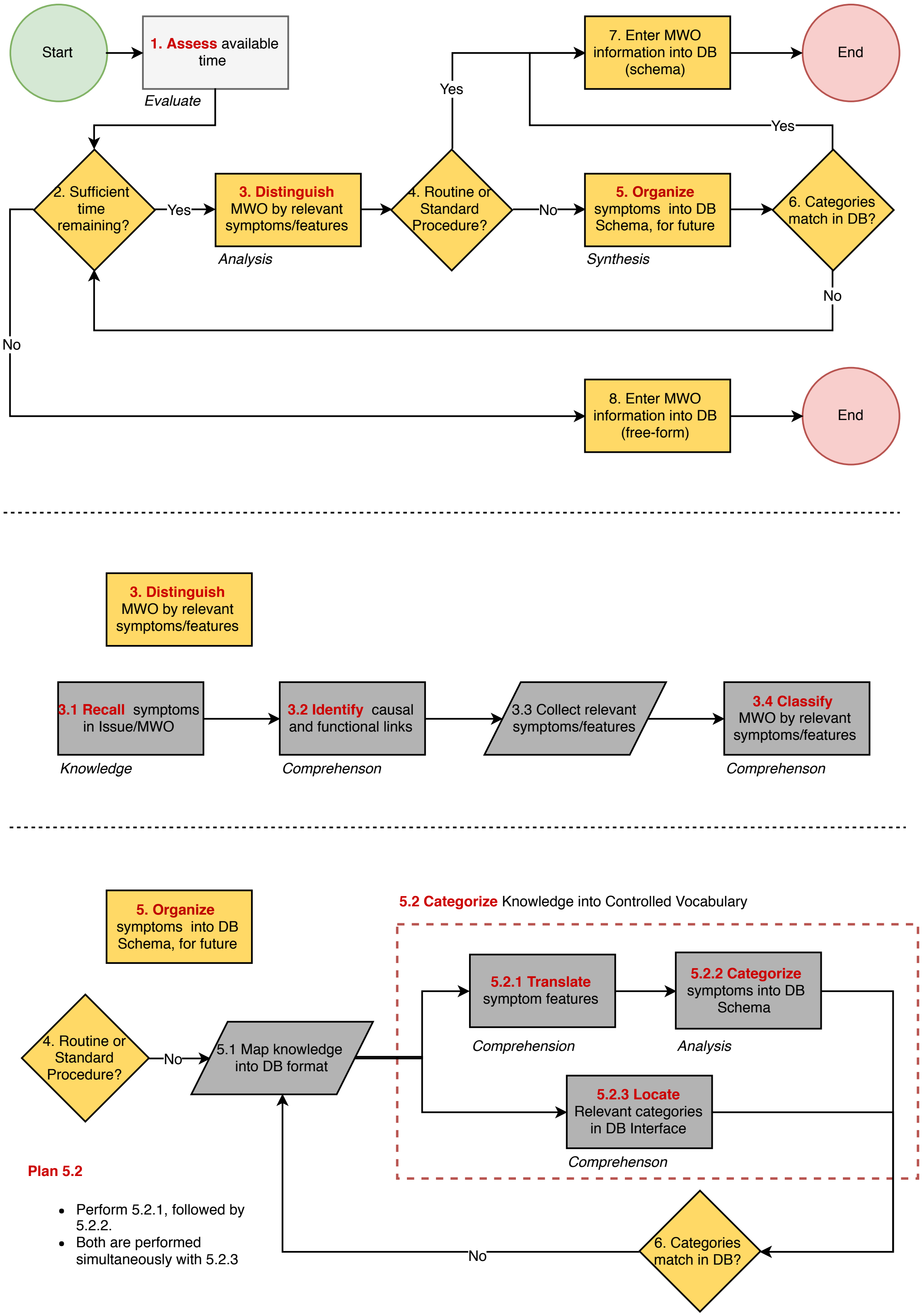
Start
1. Assess available time
Evaluate
2. Sufficient time remaining?
Yes
3. Distinguish MWO by relevant symptoms/features
Analysis
4. Routine or Standard Procedure?
Yes
No
7. Enter MWO information into DB (schema)
End
5. Organize symptoms into DB Schema, for future
Synthesis
6. Categories match in DB?
Yes
No
No
8. Enter MWO information into DB (free-form)
End
3. Distinguish MWO by relevant symptoms/features
3.1 Recall symptoms in Issue/MWO
Knowledge
3.2 Identify causal and functional links
Comprehenson
3.3 Collect relevant symptoms/features
3.4 Classify MWO by relevant symptoms/features
Comprehenson
5. Organize symptoms into DB Schema, for future
5.2 Categorize Knowledge into Controlled Vocabulary
4. Routine or Standard Procedure?
No
5.1 Map knowledge into DB format
5.2.1 Translate symptom features
Comprehension
5.2.2 Categorize symptoms into DB Schema
Analysis
5.2.3 Locate Relevant categories in DB Interface
Comprehension
Plan 5.2
Perform 5.2.1, followed by 5.2.2.
Both are performed simultaneously with 5.2.3
6. Categories match in DB?
No

Figure 1. **Top:** Cognitive Task Analysis, built on procedural tasks and Bloom's Taxonomy for congnitive levels. **Middle:** Expansion of sub-task 3. **Bottom:** Expansion of sub-task 5.

Table 2. PSF Definitions and Reference Codes

PSF	Category	PSF Code
Communication level of operator/customer to Technician	Social	A
Visibility and accessibility of system components	Technological	B
Time passed between investigation and reporting	Organization	C
Breadth of technician experience across MWO types	Personal	D
Availability and completeness of standardized procedure	Organization	E
Training in system functionality	Organization	F
Depth of technician experience in this MWO type	Technological	G
Time available for assessment	Organization	H
Technician problem-solving ability	Personal	I
Computer literacy of technician	Personal	J
Communication between management and shop-floor	Social	K
Human-system interface design	Organization	L

of structuring data-entry by utilizing only the first decision point, skipping the second entirely.

4.3. Unsafe Act Definitions and Classification

Here we enumerate the unsafe acts (UAs) associated with each task identified in Section 4, followed by the performance-shaping factors (PSFs) involved for them (refer to Table 2). We will first describe the UAs individually. This will include a proposed classification under Reasons's Error Classification (Reason, 1990) to specify whether an error is skills-based (SB), rules-based (RB), or knowledge-based (KB). The UAs are placed into a context using our PTA, followed by each PSF that context may echibit. finally, we include a table outlining the UA/PSF structure with respect to our PTA (Table 3).

Technician did not observe all symptoms relevant to the MWO (SB) (Related task: 3.1) In this situation, something prevented the technician from detecting information he needed, whether a lack of communication between the reporting operator or customer (PSF A), or a lack of accessibility and visibility into the broken system (PSF B).

Technician does not remember significant symptoms (SB) (Related task: 3.1) Depending on the amount of time that has passed between the original MWO and when the technician finally is able to monitor (PSF C), the technician may forget information that was needed. Both this and the previous UA are slips or temporary lapses.

Technician is unaware of relevant system architecture or functionality at the requisite level. (KB) (Related tasks: 3.2, 5.2.1) While identifying how recalled features of a MWO are related, or while translating these relationship into a computable format, there is a possibility that the technician is missing underlying system-level knowledge that would inform his ability to infer causality or relationships. This could be due to a lack of breadth of experience across types of machines (PSF D), the unavailability of standardized investigative procedure (PSF E), or even more generally, the amount of training the technician has received. (PSF F)

MWO has features and/or symptoms completely unlike previous experience (KB) (Related tasks: 3.4) This error manifests both when a technician has insufficient depth of experience in the specific case presented (PSF G), but also in a lack of breadth of experience to extrapolate from similar cases(PSF D). Both this and the previous UA require some level of problem solving under incomplete information, making them knowledge-based errors.

Selected MWO features are not relevant and/or MWO type is incorrectly identified (RB) (Related task: 3.4) If a technician selects MWO features, but they are not necessarily relevant, it is mostly due to the application of a heuristic or rule-of-thumb that cannot be applied here (i.e., rules-based error). This can stem from a lack of available time (PSF H), insufficient ability to problem-solve (PSF I), or again a lack of depth of experience in this type of MWO (PSF G)

Technician misunderstands target use-case and functionality of a controlled-vocabulary CMMS (KB) (Related task: 5.2.1) Here a technician must attempt to "transform" his knowledge/understanding into what he believes to be a computable format. It is possible for problems to once again stem from not only a misunderstanding of the system they maintain (UA-3), but also of the goals and functions of the computerized maintenance management system (CMMS) they are tasked with utilizing. This leads to incorrect assumptions about the type of data that is useful for later analysis, and about the level of abstraction/specificity needed in data-entry. The former is highly dependent on the communication channels being used between management and the shop-floor (PSF K), while the latter is more a function of the technician's computer literacy (PSF J)

DB schema does not contain appropriate problem code to represent the MWO features (RB) (Related task: 5.2.2) In this case, the perception that the schema is missing a problem-code is either correct, in which case it is important that the communication between management and the shop-floor is always up-to-date (PSF K), or it is wrong, and the technician may need to be more familiar with the detailed, technical behavior of the system in question (PSF G).

Technician does not apply appropriate problem code (unfamiliar with DB classification schema) (KB) (Related task: 5.2.2) It may be that the technician selects a problem code incorrectly, creating much less-useful data-set for the future. This error is occurring during an active search, making sch an error very knowledge-intensive. This is largely impacted by the training received by the technician (PSF F) and by the intuitive communication of meaning via the GUI tool (PSF L)

Technician gives up searching prior to finding appropriate problem-code. (RB) (Related task: 5.2.3) Depending on how much time is available to perform this mapping interation (PSF H) and how easy it is to search or discover classifications due to the software design (PSF L), the technician may simply give up the search for "appropriate" classifications, and default back to the closest thing he is familiar or comfortable using.

5. CASE STUDY: CREAM HRA FRAMEWORK

The above work to develop the CTA, UAs, and PSFs, is a necessary prerequisite to applying one of many available frameworks to perform HRA. Three of the most commonly used frameworks are the Nuclear Action Reliability Assessment (NARA), Standardized Plant Analysis Risk (SPAR-H), and the Cognitive Reliability and Error Analysis framework (CREAM) (Kirwan et al., 2004; Gertman et al., 2005; Hollnagel, 1998). To determine the most appropriate methodology for their own HRA case study, the National Aeronautics and Space Administration (NASA) performed a peer review in 2006, based on several selection criteria and capabilities of the various methods (Chandler et al., 2006). Given its customizability, and the generality of its foundational cognitive functions, we proceed to apply CREAM. This is intended as an example application of the task analysis constructed above, and in no way implies that CREAM is the "best" HRA framework for this task[4]. Instead, we wish to illustrate the value of HRA for systematically dissecting the complex issue of ensuring data-quality to serve smart manufacturing technologies in maintenance, by providing high-level guidance on initial risk-mitigation and further refinement in one's own operation. For a more in-depth discussion on applying CREAM to modern analyses, though in a separate domain, the reader is directed to Rashed, 2016.

CREAM, to the authors' knowledge, has not been previously used for determining HEP in MWO data-entry, and has seen little-to-no application in data-entry errors, generally. However, CREAM has been applied to estimate HEP for the shop-floor in a manufacturing setting, as seen in Schemeleva et al., 2012.

5.1. CREAM Steps

This method was outlined by (Hollnagel, 1998). As a "2nd-generation" HRA method, he proposes that all analyses must be based on some cognitive or behavioral *model* of human actions. This is what actually generates errors, and is how we have constructed Section 4. Additionally, the errors generated should be unified by a *classification* scheme, for which we have used Reason's error classification (see Table 3). Error modes are further classified by the CREAM method into three categories: Man-related, Technology-related, or Organization-related. Finally, the connection between the model and each individual error/classification must be structured via a standard *method*, which is outlined in the CREAM methodology, as summarized by NASA (Chandler et al., 2006). We certainly do not cover all of the steps involved with applying CREAM directly, but at a high level, the process consists of roughly five steps.

(1) the task sanalysis, as we have done above in a procedural task analysis. **(2)** Hollnagel, 1998 defines 15 cognitive activities to define how a human engages with each task/sub-task. Containing activities like "identify," "record," or "execute," these overlap significantly with Bloom's taxonomy, making it possible to map our proposed CTA sub-tasks to the CREAM cognitive activities, as needed (see Section 5.3. **(3)** each sub-task is assigned a human function, out of the four available (Observation, Interpretation, Planning, and Execution), that is possible to perform the cognitive activity. **(4)** each human function (O,I,P,E) can manifest errors in 2-5 specific failure modes defined by Holnagel as occurring in a function-activity combination[5], which must be mapped to our UA definitions for validation. Each valid failure mode has a so-called "basic human-error probability" (BHEP) value —these were provided by expert elicitation during CREAM's construction. **(5)** though the BHEPs set a baseline, every instance of a real system will have environmental adjustments—the PSFs— that

[4] Hollnagel states that "Although CREAM still appears to be used and referenced [...] the method from my point of view is obsolete." Newer frameworks, such as FRAM, have since been proposed (Hollnagel, 2017).

[5] For example, "O3: *observation not made*," or "E4: *Action out of sequence*. Only certain function-activity combinations are possible types of human error; e.g. the activity *Identify* can be combined with only *Interpretation* function failure modes I1: *Faulty Diagnosis*, I2: *Decision Error*, and I3: *Delayed Interpretation*

Table 3. Unsafe Act Definitions and mapping to PTA and PSFs

PTA Code	Description	Unsafe Acts	UA Code	Error Classification	PSF Codes	PSF Category
Task 3: Distingish MWO by relevant symptoms/features						
Task 3.1	Recall symptoms in issues/MWO	Technician did not observe all symptoms relevant to the MWO	UA-1	SB	A	Social
					B	Technological
		Technician does not remember significant symptoms	UA-2	SB	C	Organization
Task 3.2	Identify causal and functional links	Technician is unaware of relevant system architecture or functionality at the requisite level.	UA-3	KB	D	Personal
					E	Organization
					F	Organization
Task 3.4	Classify WO by relevant symptoms/features	MWO has features and/or symptoms completely unlike previous experience	UA-4	KB	D	Personal
					G	Technological
		Selected MWO features are not relevant and/or MWO type is incorrectly identified	UA-5	RB	H	Organization
					I	Personal
					G	Technological
Task 5: Organize symptoms into DB Schema, for future use						
Task 5.2: Categorize knowledge into controlled vocabulary						
Task 5.2.1	Translate symtpom features	Technician is unaware of relevant system architecture or functionality at the requisite level.	UA-3	KB	D	Personal
					E	Organization
					F	Organization
		Technician misunderstands target use-case and functionality of a controlled-vocabulary CMMS	UA-6	KB	J	Personal
					K	Social
Task 5.2.2	Categorize symtpoms into DB schema	DB schema does not contain appropriate problem code to represent the MWO features	UA-7	RB	G	Technological
					K	Social
		Technician does not applyappropriate problem code (unfamiliar with DB classification schema)	UA-8	KB	F	Organization
					L	Organization
Task 5.2.3	Locate relevant categories in DB interface	Technician gives up searching prior to finding appropriate problem-code.	UA-9	RB	H	Organization
					L	Organization

will modify actual HEPs. Each of these PSFs is broken down into states, such as *Available time* being "Adequate", "Temperately inadequate", or "Continuously inadequate". Each of these states, then, have corresponding weight values depending on the type of human function determined for the given sub-task. This weight is to be used in modifying the basic HEP as found above for each of the sub-tasks in our task.

With all of these in place, the formula for final HEP on a given sub-task is then

$$\hat{P}_{\text{HE}} = P_{\text{HE}}(\arg\max_{i} F_i|C_i) \times \prod_{i,j} S(Z_j(\text{PSF}_i)) \tag{1}$$

where we select the most probable failure mode F for a sub-task, given our corresponding cognitive activity C, and modify that probability by the product of scores S for each of the selected PSF$_i$, and for each of the states Z_j that PSF can occupy. The above sub-task HEP's can now be combined into a total HEP for the entire task, and specific areas can be addressed (by importance) for risk mitigation (see Section 6.1).

5.2. Possible Modifications to CREAM

Despite the overall applicability of CREAM to this HRA, several of the PSFs identified in Section 4 are not distinguished sufficiently by the generic PSFs provided in CREAM. These will rather be created, to augment the original list provided in CREAM, through a combination of data collection and statistical methods (outlined below in Section 5.3). Once new data is collected, modification values should be derived through feature-importance methods, potentially a weighted hierarchical Bayesian regression or other probabilistic model, to estimate the modifier values and uncertainties empirically. Calculation of the final HEP will then proceed as originally intended, including the new PSF values as derived *relative* to the original set.

5.3. Instantiating CREAM

As noticed in the previous section, CREAM was selected to maximize the amount of applicable generic tasks, based on a mapping between Bloom's taxonomy (as used to construct our CTA) and the list of Cognitive Activities outlined by Hollnagel, 1998. Additionally, several PSFs are well covered by the generic *Common Performance Conditions* (CPC's) he provided as well. Both sets of available data serve as a way to quickly approximate these portions of this HRA, without significant data collection, and as a way to encourage reproduceability, for which CREAM is notable (see the NASA report on CREAM (Chandler et al., 2006)).

However, 6 of the PSFs we defined contain context-specific information, with levels of important detail that the CREAM CPC defaults would approximate with too broad a brush. Three of these could be addressed through statistical analysis of existing MWO datasets—due to their importance in key sub-task locations and their ability to be approximated post-priori—and the final three could be elicited from a mixture of management and technician surveys. In addition, PSF weights must be assessed in-situ, to account for their relative importance and context sensitivity. In general, beyond preliminary guidance at PSF-to-CPC mapping, we do not proceed beyond determining BHEPs, and encourage the reader to use this work as a framework "stepping stone" toward a more complete, contextual HRA.

Table 4. Potential HRA supporting data types and sources

Data Source	Variable	Data Type
Available (CA)	T 3.1	Verify
	T 3.2	Identify
	T 3.4	Record
	T 5.2.1	Plan
	T 5.2.2	Execute
	T 5.2.3	Scan
Possible (CPC)	PSF A	Crew Collaboration
	PSF E	Procedure Availability
	PSF F	Training & Preparation
	PSF H	Available Time
	PSF K	Adequacy of Organization
	PSF L	MMI & Operational Support
Expert Elicitation	PSF B	Technician Survey
	PSF I	Management Survey
	PSF J	Management Survey
Empirical Derivation	PSF C	MWO statistical analysis
	PSF D	MWO variety metric
	PSF G	MWO recurrence metric

To maximally exploit the available data in CREAM, we briefly describe the mapping between our tasks and the CREAM Cognitive Activities (CA), through ensuring the possible types of human errors in a CA roughly correspond to identified UAs in each task.

- *3.1 Recall → Verify*: the low-level "recall" from Bloom's is not present in CREAM; however, the recall of features is a form of confirmation of state, based on prior operation, and the available O3 (*Observation not made*) and I3 (*Delayed Interpretation*) errors map well to UA-1 and UA-2, respectively.
- *3.2 Identify → Identify*: Mapping is preserved. Error I1 (*Faulty diagnosis*) covers UA-3.
- *3.4 Classify → Record*: The act of classification here is tantamount to immediate preparation for writing down verbatim whatever diagnostic features a technician has mentally developed. Note a lack of need to enter the categorization/translation loop. Execution and Interpretation errors—E5 (*Miss Action*) and I1 (*Faulty Diagnosis*), for example–could cover UA-4 and UA-5.
- *5.2.1 Translate → Plan*: The act of translation into a CMMS system requires a technician to formulate a set of actions to achieve conformity. P1 (*Priority error*) would

be indicative of a UA-3, and misunderstanding CMMS functionality (UA-6) would certainly lead to P2 (*Indadequate plan*).

- *5.2.2 Categorize → Execute*: Categorization is the final act of the technician in our HRA; he executes the translation plan while scanning for valid matches. E3 (*Action on wrong object*) covers UA-8, while E5 (*Miss Action*) could be used as a proxy for UA-7.
- *5.2.3 Locate → Scan*: Mapping is preserved. Error O3 (*Observation not made*) covers UA-9.

BHEPs can then be determined directly, using Hollnagel, 1998's estimates (See Table 5). In mapping a portion of our PSFs into the generic CREAM CPC's, we limit these equivalences to only those cases where a single PSF is sufficiently covered by a single CPC in an obvious way. These mappings can be found in Table 4. These are only possibilities, and must still be scored and combined with the other context-specific PSFs to create a complete HRA. We leave a discussion of expert-derived or data-driven estimation for these PSFs to potential future case studies.

6. DISCUSSION

Upon construction and execution of the HRA method, an analyst may recommend strategies for risk mitigation, along with understanding any implications that performing this HRA might imply. We attempt to foresee several major areas that such a study might address.

6.1. Risk Mitigation

The final step in a fully realized HRA would be to calculate context-specific PSF values, which can directly provide the HEP via Equation 1. As such, they cannot be provided here, and require additional data collection and user study as outlined in Table 4. Still, the BHEP values as determined by applying CREAM to our CTA can provide valuable insight into possible *categories* of failure modes and risk mitigation strategies to be aware of, going forward.

Based on the CREAM function-activity mappings, which were arrived at through systematic mapping to our unsafe acts via Section 5.3 and summarized in Table 5, two main risk ”areas” dominate the BHEP values. **Interpretation** errors (mostly I1: *Faulty Diagnosis*) leading up to the act of data-entry (related to a technician's understanding of the asset/system, UA-3), and **Planning/Execution** during the data-entry categorization loop (P2, E5, related to UA-6 and UA-7). Though the table outlines a more complete breakdown by individual sub-task, these failure modes will likely take mitigation priority moving forward.

In the spirit of avoiding personal blame during error mitigation in favor of systemic change (Reason, 1990), strategies must be designed and executed with full understanding of each organizational context. The suggested risk mitigation strategies in Table 5, then, should be viewed merely as starting points in a more holistic apporach, guided by the framework we provided above. In the interest of guidance for the readuer, we have categorized the suggested mitigation strategies into four archtypes — Control (legistlation, standards, etc.), Educate, Design (system/interface, engineering principals, etc.), and Persuade.

The approaches outlined in Table 5 generally fall into two categories: 1) modify technician behavior through education and environment adjustments to reduce errors in diagnosis, and 2) design a data-entry framework that minimizes execution and planning error through cognitive analysis and good HMI principles.

Training and diagnostics First, as diagnostics is the primary job of a maintenance technician, and the cost of error is already known to be high, it is unlikely that the mitigation strategies presented here will significantly impact whatever mitigation is already in place through a maintenance department. The scope of this HRA necessarily includes the diagnostic capabilities of a technician as a *prior* to MWO data entry, but actual alteration of the course of normal maintenance duties is outside the scope of feasible mitigation. As such, we outline only relatively low-cost strategies that are industry best-practice already. Examples given are: formalization of asset/system architectures for easy reference and guidance, institution of a buddy-system for new hires or assessment of new asset types, and institutional requirement of periodic data-entry times to reduce stress from time constraints. In short, any risk met here should necessarily be addressed by the maintenance team in the course of their duties, regardless of a data-entry component.

HMI and data-entry design This point in the decision process is much closer to the original intent of this HRA. Risks are primarily found in the disconnect between the needs of a structured database (its abstraction level, ability to query past events, modeling of asset/system architecture, etc.) and the needs of a technician (pressure to be thorough vs timely, use of efficient jargon, reliance on tacit knowledge through expertise). These can be mitigated, at relatively low cost, depending on the level of existing data entry adoption.

Already using controlled vocabulary — If an organization is already tied to a particular DB architecture, it is possible to illicit technician feedback on pain points through current usage patterns to either better communicate the level of detail needed for future operational usage of the database (low-cost), or design a more functional and user-driven interface through HMI principals (medium-high cost), or—preferrably—both. Failure to do so will likely result in unwanted data-entry behavior indefinitely, as a feeling of resent-

Table 5. Risk Mitigation Strategies, by Unsafe Act

Task	UA→CA	BHEP	Risk Mitigation
T 3.1	UA-1→O3	3E−3	**Persuade** - promote positive interaction of operator and tecnhician
	UA-2→I3	1E−2	**Control/Design** - designate time as set aside for data-entry throughout the day
T 3.2	UA-3→I1	2E−1	**Educate** - provide high and low-level system architecture as reference material at diagnosis-time
T 3.4	UA-4→E5	3E−2	**Educate/Control** - institute ”buddy” system for newly aquired assets and new hires
	UA-5→I1	2E−1	**Persuade** - motivate technicians to expound upon diagnostic decisions in MWO, and add more relevant details when unsure.
T 5.2.1	UA-3→I1	2E−1	**Educate** - provide high and low-level system architecture as reference material at diagnosis-time
	UA-6→P2	1E−2	**Design/Educate** - make use of interface that obviates correct usage and learning curve through HMI design principles.
T 5.2.2	UA-7→E5	3E−2	**Design** - only require specific, controlled data-entry at the necessary abstraction level
	UA-8→E3	5E−4	**Design** - incorporate feedback from technicians on interface design
			Educate/Control - institute ”buddy system” for newly aquired assets and new hires.
T 5.2.3	UA-9→O3	3E−3	**Design/Persuade** - ensure data-entry attitude toward thouroughness matches the requisite abstraction and detail level of DB

ment could build between data-management personnel and shop-floor technicians (due to perpendicular goals).

Implementation of new system — preliminary research indicates that the usage of a more user-driven data-structuring approach could arrive at a desired level of abstraction *naturally*, through use of tags, for example (Sexton et al., 2017; Sinha, 2005). This also implies the direct usage of existing technician shorthand, via data-driven vocabulary mappings. Such a system is low-cost, but may require significant re-training of multiple parties in using such a new paradigm. Further research is needed; however, there is potential to completely mitigate the otherwise unavoidable risk associated with entering the categorization loop in Task 5.2

6.2. Implications for HRA adoption

The proposed HRA method has been developed to minimize the data collection and computational work-load required for estimation of HEP for two major tasks in MWO data-entry. Stakeholders, such as SME manufacturers and other industrial maintenance operations, or operations researchers, have influenced its development through feedback on major problems encountered when attempting to utilize existing MWO data. As such, if synthesis of these HRA results continue to be broken down into guidelines that are dependent on the current stage of data-entry implementation, acceptance is potentially straightforward across many organizational scales, *by design*.

It is, however, necessary that the cultural interest of an organization be aligned with the implementation of more data-driven “smart manufacturing” methodologies. It would be potentially beneficial to assess the maturity of an organization for implementing such technologies prior to undertaking or utilizing results from the HRA method proposed here, and several tools already exist or are in development to do so (De Carolis et al., 2017).

6.3. Limitations

It is important to keep several limitations of the methodology proposed here in mind, moving forward. First are limitations surrounding CREAM itself; despite being a 2$^{\text{nd}}$ generation HRA method—being more grounded in underlying human cognition than previous schemes—it has not been designed for this domain, nor validated formally in it. This can be seen in the lack of specificity in several mappings between our UAs and the CREAM Cognitive Activity/Human Function pairings. These mappings will need to be validated numerically as useful, before they should significantly impact policy of any kind. The same is true for the base-values for HEP given, which were not only designed for a different domain, but were originally a mixture of values from previous HRA methods and elicitation from experts (Chandler et al., 2006).

Another key limitation is in the ability for useful PSF values to be estimated from existing MWO datasets, which are a very common existing data-type for maintenance operations. The size of these datasets, and their structuring quality, is severely lacking for highly-dimensional statistical models necessary—thus the existence of this HRA in the first place. It is not yet known if results across organizations and domains (which will be needed to obtain sufficient data) will be compatible, nor to what degree the subjectivity differences between management and technician surveys will affect PSF values. This must be kept as a high-priority discussion point in any implementation of this HRA.

7. CONCLUSION

We have proposed a HRA for the quantification and potential mitigation of human error in MWO data entry through categorization, with the goal of increasing the reliability of these human-machine systems. This is done by assessing the components of a proposed two-part process humans use to structure data: determine distinguishing features of a given datapoint, and organize those features into the database schema for future queries. Using principals of cognition, behavior, and risk assessment, it is hoped that the sources of risk in each can be mitigated at low-cost through implementing education, design, and socially-driven strategies, along with a re-thinking of the current MWO paradigms to incorporate human-in-the-loop data-driven pipelines that take advantage of technicians' expertise, rather than antagonize it.

Future extensions of this HRA should perform similar analyses for the other two identified forms of data-entry, followed by a decision guidance study that weighs the calculated risks of each data-entry system with the costs associated in their usage. Preliminary research indicates potential usefulness of hybridized data-structuring systems that could more efficiently balance these two driving forces.

DISCLAIMER

The use of any products described in this paper does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that products are necessarily the best available for the purpose.

REFERENCES

Adams, N. E. (2015). Bloom's taxonomy of cognitive learning objectives. *Journal of the Medical Library Association: JMLA*, *103*(3), 152.

Brundage, M. P., Kulvatunyou, B., Ademujimi, T., & Rakshith, B. (2017). Smart manufacturing through a framework for a knowledge-based diagnosis system. In *ASME 2017 12th international manufacturing science and engineering conference* (pp. V003T04A012–V003T04A012).

Chandler, F., Chang, Y., Mosleh, A., Marble, J., Boring, R., & Gertman, D. (2006). Human reliability analysis methods: selection guidance for NASA. *NASA Office of Safety and Mission Assurance, Washington, DC, 123*.

De Carolis, A., Macchi, M., Kulvatunyou, B., Brundage, M. P., & Terzi, S. (2017). Maturity models and tools for enabling smart manufacturing systems: comparison and reflections for future developments. In *Ifip international conference on product lifecycle management* (pp. 23–35).

Fazio, R. H., Williams, C. J., & Powell, M. C. (2000). Measuring associative strength: Category-item associations and their activation from memory. *Political Psychology*, *21*(1), 7–25.

Gertman, D., Blackman, H., Marble, J., Byers, J., Smith, C., et al. (2005). The spar-h human reliability analysis method. *US Nuclear Regulatory Commission*.

Heinze, D. T., Morsch, M. L., & Holbrook, J. (2001). Mining free-text medical records. In *Proceedings of the amia symposium* (p. 254).

Hollnagel, E. (1998). *Cognitive reliability and error analysis method (CREAM)*. Elsevier.

Hollnagel, E. (2017). *Fram: the functional resonance analysis method: modelling complex socio-technical systems*. CRC Press.

Kegg, R. L. (1984). One-line machine and process diagnostics. *CIRP Annals - Manufacturing Technology*, *33*(2), 469 - 473. Retrieved from `http://www.sciencedirect.com/science/article/pii/S0007850616301688` doi: http://dx.doi.org/10.1016/S0007-8506(16)30168-8

Kirwan, B., Gibson, H., Kennedy, R., Edmunds, J., Cooksley, G., & Umbers, I. (2004). Nuclear action reliability assessment (nara): a data-based hra tool. In *Probabilistic safety assessment and management* (pp. 1206–1211).

Loch, F., & Vogel-Heuser, B. (2017). A virtual training system for aging employees in machine operation. In *15th ieee international conference on industrial informatics (indin)*.

Logan, G. D. (2004). Cumulative progress in formal theories of attention. *Annu. Rev. Psychol.*, *55*, 207–234.

Luce, R. D. (1977). The choice axiom after twenty years. *Journal of mathematical psychology*, *15*(3), 215–233.

Molina, R., Unsworth, K., Hodkiewicz, M., & Adriasola, E. (2013). Are managerial pressure, technological control and intrinsic motivation effective in improving data quality? *Reliability Engineering & System Safety*, *119*, 26–34.

Pantförder, D., Schaupp, J., & Vogel-Heuser, B. (2017). Making implicit knowledge explicit–acquisition of plant staff's mental models as a basis for developing a decision support system. In *International conference on human-computer interaction* (pp. 358–365).

Peters, I. (2009). Folksonomies. indexing and retrieval in web 2.0. In (pp. 162–164). Walter de Gruyter.

Rashed, C. S. K. (2016). The concept of human reliability assessment tool CREAM and its suitability for shipboard operations safety. *Journal of Shipping and Ocean Engineering*, *6*, 348–355.

Rasmussen, J. (1983). Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE transactions on systems, man, and cybernetics*(3), 257–266.

Reason, J. (1990). *Human error*. Cambridge university press.

Schemeleva, K., Nguyen, C., Durieux, S., & Caux, C. (2012). Human error probabilities computation for manufacturing system simulation using CREAM. In *9th international conference on modeling, optimization & simulation.*

Sexton, T., Brundage, M. P., Hoffman, M., & Morris, K. C. (2017, Dec). Hybrid datafication of maintenance logs from ai-assisted human tags. In *2017 ieee international conference on big data (big data)* (p. 1769-1777). doi: 10.1109/BigData.2017.8258120

Sinha, R. (2005). A cognitive analysis of tagging (or how the lower cognitive cost of tagging makes it popular). *Rashmi Simha: thoughts on technology, design & cognition*.

Thomas, D. S. (2018). *The costs and benefits of advanced maintenance in manufacturing* (Tech. Rep.).

Thurstone, L. L. (1927). A law of comparative judgment. *Psychological review*, *34*(4), 273.

Tremblay, M. C., Berndt, D. J., Luther, S. L., Foulis, P. R., & French, D. D. (2009). Identifying fall-related injuries: Text mining the electronic medical record. *Information Technology and Management*, *10*(4), 253.

Unsworth, K., Adriasola, E., Johnston-Billings, A., Dmitrieva, A., & Hodkiewicz, M. (2011). Goal hierarchy: Improving asset data quality by improving motivation. *Reliability Engineering & System Safety*, *96*(11), 1474–1481.

Zhou, X., Han, H., Chankai, I., Prestrud, A., & Brooks, A. (2006). Approaches to text mining for clinical medical records. In *Proceedings of the 2006 acm symposium on applied computing* (pp. 235–239).