EPRI

Automating Corrective Action Programs in the Nuclear Industry

2022 TECHNICAL REPORT

Automating Corrective Action Programs in the Nuclear Industry

All or a portion of the requirements of the EPRI Nuclear Quality Assurance Program apply to this product.

YES  NO

EPRI Project Manager
C. Wiegand

3420 Hillview Avenue
Palo Alto, CA 94304-1338
USA

PO Box 10412
Palo Alto, CA 94303-0813
USA

800.313.3774
650.855.2121

askepri@epri.com

www.epri.com

3002023821
Final Report, June 2022

DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITIES

THE FOLLOWING ORGANIZATION, UNDER CONTRACT TO EPRI, PREPARED THIS REPORT:

Nuclearn, Inc.

THE TECHNICAL CONTENTS OF THIS PRODUCT WERE **NOT** PREPARED IN ACCORDANCE WITH THE EPRI QUALITY PROGRAM MANUAL THAT FULFILLS THE REQUIREMENTS OF 10 CFR 50, APPENDIX B. THIS PRODUCT IS **NOT** SUBJECT TO THE REQUIREMENTS OF 10 CFR PART 21.

NOTE

For further information about EPRI, call the EPRI Customer Assistance Center at 800.313.3774 or e-mail askepri@epri.com.

Together...Shaping the Future of Energy®

Acknowledgments

The following organization, under contract to the Electric Power Research Institute (EPRI), prepared this report:

Nuclearn, Inc.
4848 E. Cactus Road
Suite 505-1814
Scottsdale, AZ 85254

Principal Investigators
B. Fox
J. Vincent

This report describes research sponsored by EPRI.

EPRI thanks the following people and organizations for their contributions to this report:

A. Y. Al Rashdan	Idaho National Laboratory
P. Byers	Xcel Energy
G. Kelly	Exelon Corporation
S. Lappegaard	Xcel Energy
J. Rigatti	Dominion Energy
J. Slider	Nuclear Energy Institute
B. Wright	Arizona Public Service

This publication is a corporate document that should be cited in the literature in the following manner:

Automating Corrective Action Programs in the Nuclear Industry. EPRI, Palo Alto, CA: 2022. 3002023821.

Abstract

Nuclear corrective action programs (CAPs) are a foundational block of a nuclear safety culture, providing the implementation and execution process for problem identification, resolution, and continuous learning for the nuclear industry. But with that importance comes a heavy burden in the way of personnel resources resulting from the manual review and the screening process that is currently employed by most utilities.

This report provides guidance for a utility in implementing automation into its CAP. The report provides an overview of CAP processes that can be automated, fundamental techniques for automation, and key considerations for adopting an automated CAP system.

Keywords
Artificial intelligence
Corrective action program (CAP)
Machine learning
Natural language processing

Deliverable Number: 3002023821

Product Type: Technical Report

Product Title: Automating Corrective Action Programs in the Nuclear Industry

PRIMARY AUDIENCE: Nuclear utility innovation managers, performance improvement managers, corrective action program (CAP) managers

SECONDARY AUDIENCE: Nuclear utility corporate and site senior leaders

KEY RESEARCH QUESTION

Can the CAP review process be automated? This question prompts an investigation into whether the burden associated with the CAP program can be reduced by automation, what the solution consists of, and the current adoption status of such a solution within the nuclear industry.

RESEARCH OVERVIEW

This research used experience and knowledge in implementing automation into a CAP at a nuclear power plant (NPP) and compiled operating experience from several NPPs in implementation and usage of artificial intelligence and machine learning techniques. The research also introduces the field of artificial intelligence and machine learning to members of the nuclear industry who may have not had any interaction with those technologies.

KEY FINDINGS

- The automation process involves engaging artificial intelligence techniques, which is a relatively new concept for the nuclear industry; so, a brief introduction is included in the report.
- Benefits of an automated CAP system (ACAPS) are described and include the following:
  - Reduction in labor hours spent supporting CAP processes
  - Decreased lead time in CAP processes
  - Improved consistency of CAP processes
  - Ability to trend and analyze historical data in ways that were previously cost-prohibitive

WHY THIS MATTERS

Nuclear CAPs are a foundational block of a nuclear safety culture, providing the implementation and execution process for problem identification, resolution, and continuous learning for the nuclear industry. But with that importance comes a heavy burden in the way of personnel resources resulting from the manual review and the screening process that is currently employed by most utilities. This report provides guidance for a utility in implementing automation into its CAP.

HOW TO APPLY RESULTS

After reviewing this report, readers should have the knowledge necessary to organize efforts to adopt CAP automation and should be better able to evaluate the potential success of various in-house, vendor, and consulting approaches.

LEARNING AND ENGAGEMENT OPPORTUNITIES

Other organizations with direct or peripheral interests, such as the Corrective Action Program Owners Group and Nuclear Energy Institute's Innovation working group, will find this report useful.

EPRI CONTACT: Christopher Wiegand, Senior Technical Executive, cwiegand@epri.com

PROGRAM: AI.EPRI

IMPLEMENTATION CATEGORY: Reference

EPRI
3420 Hillview Avenue, Palo Alto, California 94304-1338 • PO Box 10412, Palo Alto, California 94303-0813 USA
800.313.3774 • 650.855.2121 • askepri@epri.com • www.epri.com

Acronyms and Definitions

Acronyms

ACAPS	automated corrective action program system
AI	artificial intelligence
API	application programming interface
AUC	area under curve
CAP	corrective action program
CAQ	condition adverse to quality
CR	condition report
CRG	condition review group
DOE	U.S. Department of Energy
EAM	enterprise asset management
ELK	Elastic, Logstash, Kibana
EPRI	Electric Power Research Institute
FLM	front line manager
INPO	Institute of Nuclear Power Operations
IT	information technology
KNN	K-nearest neighbor
LDA	latent Dirichlet allocation
ML	machine learning
MRFF	Maintenance Rule functional failure
NLP	natural language processing
NPP	nuclear power plant
NRC	Nuclear Regulatory Commission
SCAQ	significant condition adverse to quality
SVM	support vector machine
WANO	World Association of Nuclear Operators
$	United States dollar

Definitions

CAPs across different NPPs have varying terminology for referring to CAP components. To simplify the remainder of this report, the following terminology will be used to refer to CAP components:

CAP source system. The computer system that retains records of condition reports, corrective actions, and related information.

Condition adverse to quality (CAQ). A deviation from a requirement, a deficiency, or some other condition that could adversely impact public or personnel health and safety, waste acceptance, the environment, facility operations, or the effective implementation of the quality assurance program.[1]

Condition report (CR). A submitted report that documents a potential condition adverse to quality in or around a nuclear power plant.

Corrective actions. Actions issued to other groups that evaluate, correct, or further document the CR.

CR initiation. The act of writing a CR after becoming aware of a potential condition adverse to quality.

Front line manager (FLM) review. Review of a CR by the initiator's first-level leadership. Usually intended to ensure quality, accuracy, and severity.

Level of effort. Amount of effort and depth of investigation expected to be expended to investigate and correct an identified condition or its causes.

Management review group review. Review of CRs conducted by a group of mid- to senior-level managers for quality, accuracy, and severity, as well as priority and trending. Typically performed as a step in the middle or end of the condition review process.

---

[1] Palay, Christian, Preparer; Murray, Robert, Director; U.S. Department of Energy Office of Standards and Quality Assurance; Administrative Procedure AP-16.1Q Rev 1, Subject: "Corrective Action," 2011. https://www.emcbc.doe.gov/Content/Office/ap_16_1q_rev_1_corrective_action_07_15_11.pdf.

Responsible group. Refers to either the owning unit of the overall CR responsible for overseeing resolution or the owner of a corrective action.

Screening committee. A group of individuals who review CRs for various fields (significance, ownership, and so on) and distribute actions to groups for correction. Can be a dedicated collective of individuals or a distributed effort shared among many.

Significance. Level of severity of the report capturing the impact to a nuclear power plant's reliability, safety, generation potential, or regulatory standing.

The following are machine learning—or artificial intelligence—specific terms defined for consistency between practitioners when referencing this report:

Artificial intelligence (AI). AI is a system or systems that can intake a variety of unfiltered information of different modes, analyze that information, and produce a variety of decisions based on intricate patterns contained within that information.

Category/class/field type. The name of a differentiating detail between otherwise common objects.

Concept drift. The slow changing in the relationship between input data and target variable over time.[2]

Decision boundary. The region in feature space where a class label decision is ambiguous. May be a point, plane, or hyperplane in which target class labels become separable.

False positive. A prediction made by a predictor that is said to be true but is actually false.

False negative. A prediction made by a predictor that is said to be false but is actually true.

Feature importance. The measurement of how influential a single feature is when a model produces a prediction.[3]

---

[2] Gama et al., "A Survey on Concept Drift Adaptation," 2013, http://eprints.bournemouth.ac.uk/22491/1/ACM%20computing%20surveys.pdf.

[3] Casalicchio et al., "Visualizing the Feature Importance for Black Box Models," 2018, https://arxiv.org/pdf/1804.06620.pdf.

Machine learning (ML). The ability of a machine or computer program to seemingly learn, or associate inputs to outputs, from the outcome of a recorded decision without predefined rules or logic.

Natural language processing (NLP). A subfield of AI focused on the interactions of computing and human language, specifically programming computers to understand natural human language.

Online learning. The act of learning from training data records one-by-one as they are encountered versus batch or offline learning in which training records are processed in bulk.[4]

Vocabulary. The unique set of terms in a language; the simplest repeating element in a language.

[4] Hoi et al., "Online Learning: A Comprehensive Survey," 2018, https://arxiv.org/pdf/1802.02871.pdf.

Legend

This report blends together a corrective action program automation how-to guide; relevant operating experience from the nuclear industry; technical information from the fields of software, data science, and artificial intelligence; identified best practices from executed projects; and important points. To better identify these areas, the sections have been highlighted in the following colors:

Key Operating Experience
Key Important Points
Key Technical Information
Key Cost/Value Considerations
Key Identified Best Practices

Also note that wherever the symbol *$* is used, it is indicating values in U.S. dollars at the average 2021 value.

Table of Contents

List of Figures

List of Tables

Section 1: Introduction

Nuclear corrective action programs (CAPs) are a foundational block of a nuclear safety culture, providing the implementation and execution process for problem identification, resolution, and continuous learning for the nuclear industry. But with that importance comes a heavy burden in the way of personnel resources resulting from the manual review and screening process that is currently employed by most utilities.

This report provides guidance for a utility in implementing automation into its CAP. The report provides an overview of CAP processes that can be automated, fundamental techniques for automation, and key considerations for adopting an automated correction action program system (ACAPS).

The automation process involves engaging artificial intelligence (AI) techniques, which is a relatively new concept for the nuclear industry; so, a brief introduction is included in the report.

Operating experience from nuclear power plants (NPPs) that have already initiated CAP automation projects or other process automation projects is also included in the report. Note that NPP/utility names have been anonymized; interested readers can contact the Electric Power Research Institute (EPRI) project manager for additional information.

After reviewing this report, readers should have the knowledge necessary to organize efforts to adopt CAP automation and should be better able to evaluate the potential success of various in-house, vendor, and consulting approaches.

The goals of CAP automation will vary for each NPP, but they primarily consist of realizing the benefits of eliminating a portion of manual CAP processes by automating repetitive tasks and decisions using an ACAPS. These benefits include the following:

- Reduction in labor hours spent supporting CAP processes
- Decreased lead time in CAP processes (automated records are processed almost instantly)
- Improved consistency of CAP processes
- Ability to trend and analyze historical data in ways that were previously cost-prohibitive (apply trend codes to historical data)

To maximize these benefits, an ACAPS should aim to meet the following goals:

- Automate key decisions and tasks with a high degree of accuracy
- Be able to differentiate between decisions it can make with high confidence that can be automated and those that are less confident and should be manually processed
- Integrate directly with the CAP source system workflow

As the nuclear industry looks to find more efficient ways of doing business while maintaining or improving the level of safety, utilities interested in implementing CAP automation will find this report useful for relevant operating experience, best practices, and common pitfalls.

Section 2: AI Introduction

Before understanding how an ACAPS functions, it is important to have a rudimentary understanding of AI concepts that relate to CAP automation.

Overview

Fundamentally, AI is the ability of machines to recognize incoming information, make decisions based on that information to achieve a desired outcome, and learn from the outcome of those decisions so that future decisions are more likely to be desirable. The practice of AI involves building machines, systems, and/or computer programs that can achieve this pattern of operation. Although AI systems can be simple, most commonly AI is a composition of systems able to intake data, use machine learning (ML) to associate patterns in these data with desired outputs, and produce those outputs when required.

The enabling piece of these systems, which both analyzes incoming information and learns from the outcome of the decision made, is implemented through the concept of ML. ML is the ability for a machine or program to seemingly learn from the outcome of a recorded decision. The types of decisions can be simple, such as selecting *Yes* or *No*. Or they can be complex, such as selecting the next word in a passage of text from the entire English language or determining the steering angle in a self-driving car. *Learning* in the context of ML is the cycle of asking the program to make a decision based on some set of information, recording the outcome of the decision, and providing feedback to the program so that the outcome of the current decision is incorporated into the next decision. At a very high level of understanding, the process of training a machine program to make the expected decision is not unlike teaching a dog how to perform a new action: practice cycles and appropriate feedback cause positive changes in future actions.

Modern AI Functions

There are many branches of AI, including reasoning, problem solving, knowledge representation, planning, learning, perception, and general intelligence. In modern AI applications, computer algorithms are developed and used to demonstrate intelligence with varying degrees of

human involvement in the development, configuration, and use of these AI algorithms. Of the different branches of AI, ML and probabilistic reasoning are the most relevant to CAP automation.

Machine Learning

ML is the process of a computer program or algorithm adapting itself to produce a desired outcome without input from a human agent. ML is how humans teach a computer program to recognize input CAP data, such as a condition report (CR) description and author name, and associate various patterns in those inputs to the desired output value.

Types of ML

Within ML, there are two primary types of models: supervised and unsupervised. The primary difference between supervised and unsupervised models is that supervised models train on labeled data sets and are generally used to predict an output based on a set of inputs. Unsupervised models are trained on unlabeled data sets and learn patterns and information about the data sets themselves. Unsupervised models are often later used to improve supervised models, such as contextual embeddings[5] being used for natural language processing (NLP) classification tasks. Another use of unsupervised models is for anomaly detection—that is, detecting when new inputs are very different from the inputs in the training data.

Supervised models can be used to predict a variety of different outputs but, in the context of CAP automation, are typically used for classification tasks. Classification tasks are problems that can be solved by supervised ML models learning to predict one or more labels for a given input. Common types of classification tasks include the following:

- **Binary classification**. Classification model predicts whether a given record belongs to one class. This is the ML equivalent of a *Yes or No* question.
- **Multiclass classification**. Classification model predicts which class a given record belongs to when there are multiple classes available. This is the ML equivalent of a *Choose A, B, C, or D* question.
- **Multilabel classification**. Classification model predicts which labels apply to a given record, and anywhere between none and all of the labels can apply. This is the ML equivalent of a *Select all correct answers* question.

[5] See Understanding Words Using Context section of this report.

Probabilistic Reasoning

Probabilistic reasoning is the ability to quantify uncertainty with an associated outcome and incorporate that uncertainty into future decision making. In CAP automation, probabilistic reasoning is used to determine whether a CR should be processed for automation given an automation model output confidence level. References to setting thresholds to model output probabilities or confidence levels are examples of probabilistic reasoning.

Teaching AI to Understand Nuclear Language

Because much of the information available in a CAP is provided by free-form text fields, one of the fundamental challenges in CAP automation involves teaching ML algorithms how to best understand and interpret nuclear-specific, natural language input. How does one teach a machine that operates in 1s and 0s how to understand nuclear language? Most ML techniques convert the base common components of the language (for example, a word or phrase) into a number and then teach the computer to understand relationships between those numbers. The exact methods for accomplishing this are varied and ever-evolving as research becomes accepted practice and the computing continues on a downward cost trend. The selection of the best methods to use depends on several factors: the level of expertise of the practitioner, the amount of language data available, the variety of the language, the amount of specialized computational resources available, and the desired level of language modeling that is actually needed to accomplish the end task.

Bag of Words and Term Counts

As a simple example, consider the following sentence:

> On 1/1/2020, the solenoid valve failed stuck open and thus failed the stroke closed test.

A traditional method for converting these data into something that a machine can detect patterns in involves counting the number of times specific terms appear in the text. In the preceding example, *solenoid* and *valve* appear once each. *Failed* appears twice. Based on these counts, a machine could likely determine that this sentence could be bucketed into an equipment failure CR bucket. This is an example in which words are converted into counts (numbers), and counts are used to predict the category of the CR (relationships between words are discarded) (see Figure 2-1).

Term	On	1/1/2020,	the	solenoid	valve	failed	stuck	open	and	thus	stroke	closed	test
Count	1	1	2	1	1	2	1	1	1	1	1	1	1

Figure 2-1
Example text term frequencies

Although this approach is relatively easy to use, it removes a lot of information contained in the sentence, particularly the order of the words themselves. In addition, nothing is learned about the words themselves—they are simply converted to a bucket for counting.

Understanding Words Using Context

A common modern approach uses large neural networks to tie all the words together when making a prediction. To understand this approach, one must understand how humans are able to use context to infer the meaning of a word or even what words make sense to use in a sentence. Attempt to predict the {HIDDEN} word in the following sentence:

> On 1/1/2020, the auxiliary feedwater {HIDDEN} was discovered to have an oil leak when started for a condensate water transfer operation.

As one attempts to guess the {HIDDEN} word, both the leading and trailing contexts are used. Someone familiar with nuclear power would likely guess that the following set of words—{*pump*, *seal*, *bearing*}—is likely to take the place of {HIDDEN} and would also know that a word such as *dog* or *software* would be unlikely. Beyond just the neighboring words such as *auxiliary*, *oil leak*, and *condensate* being present, one can infer a narrow set of words just from the structure of the sentence. Humans can do this because many examples of similar sentences or phrases have been learned from the relationships between specific words and the context in which they are used.

Many modern ML algorithms are taught by inverting this example. They will read millions of sentences in a particular language, artificially censor specific words (as in the preceding example), and learn the patterns of what words should appear. The algorithm eventually learns that words such as *pump*, *seal*, and *bearing* are often used in very similar contexts. These words are then assigned numeric values that are very close to one another because they are used in the same context. Unrelated terms, such as *dog* or *software*, are assigned numbers much further away, because they are highly unlikely to be used in the same context as *pump*, *seal*, or *bearing*[6] (see Figure 2-2).

[6] This is often done in higher dimensional space where, instead of a single number, one can randomly select a vector of numbers to represent the word.

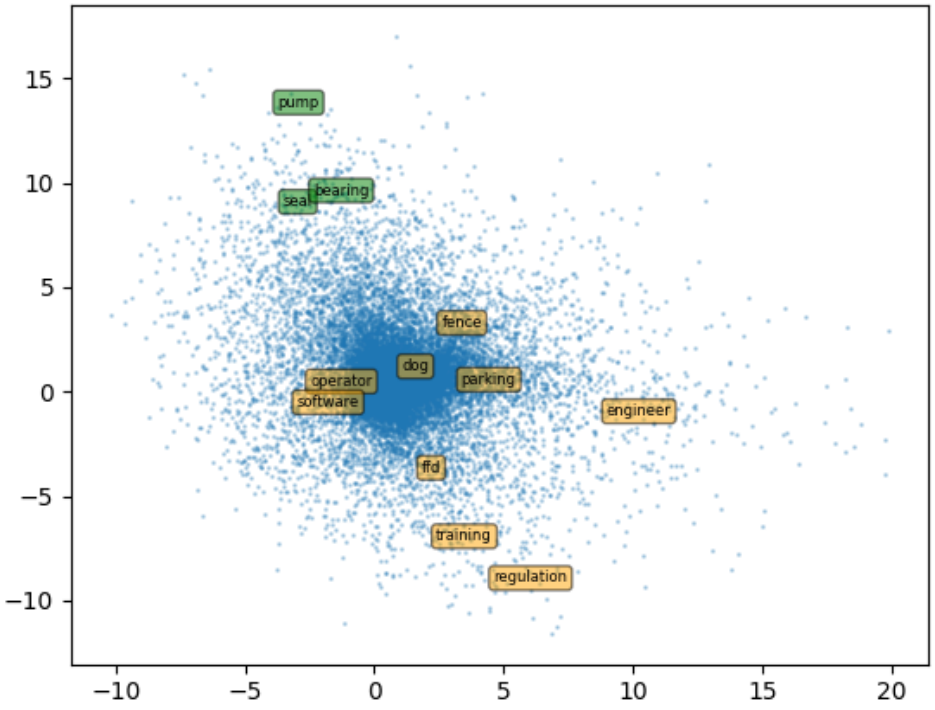


Figure 2-2
Principal component analysis of Word2Vec embeddings in nuclear domains

If this process is repeated millions of times on tens of thousands of words, an ML algorithm can map each term in a vocabulary to a location on a number line. This numeric representation captures a lot more than just the presence of a term in a sentence; it can actually represent the concept or meaning of the term itself. This number can then be directly used by ML algorithms and has been shown to provide dramatic improvements in accuracy in many different problem domains.

Full Context

The absolute state-of-the-art ML algorithms take this process a step further. The context around a word can also change the meaning of a word. In addition, some words or phrases can be much more important for understanding a sentence than others. The latest ML techniques using large language models are actually able to learn how to alter the numeric representation of terms by evaluating the context of the sentence as well as increase or decrease the weight of these words in an ML task based on how important they may be. These algorithms are complex and require large amounts of data and processing power to train but deliver significantly more accurate results for many tasks.

Applications to Nuclear

Although many of these methods are used for building ML models that understand the English language, these same methods can be used to build models tailored to understand the intricacies of specific domains. As it relates to the subject of this report, models can be trained using only nuclear-specific data or even CAP-specific data. These models are then able to learn how to understand terms as they are used in nuclear or even learn terms that only appear in nuclear-specific data.

A summary of the techniques covered in this section appears in Table 2-1.

Table 2-1
Nuclear language modeling techniques

Method	Description	Category	Difficulty	Pros	Cons
Bag of words	Uses the number of appearances of a word in a passage to characterize the passage.	Count based	Easy	• Simple • Fast • Cheap	• Limited performance. • Information lost in positioning and relationship. • Creates high dimensional data sets.
Term frequency–inverse document frequency	Uses the frequency of word appearances in a passage combined with how often a word exists in all passages to characterize a passage. Used to boost the relevancy of rare words in numerically describing the passage.	Count based	Easy	• Fast • Cheap • Improved recognition and impact of rare words	• Limited performance. • Lost information in positioning. • Creates high dimensional data sets.
Word embeddings (Word2Vec/GloVe)	A set of numbers (vector) created for each word in a language, each adjusted to represent a word based on nearby occurrences of other words.	Similarity vectors	Medium	• Fast • Context aware • Greater expressivity • Lower dimensionality	• Representations do not differ with changes in context. • Need to precompute prior to utilization.
Recurrent neural networks	A neural network architecture trained in such a way that output is used as input in a recursive fashion. Creates a numerical representation of a sequence of words.	Context sensitive	Medium	• Context aware • Excellent expressivity	• High resource utilization. • Slow to develop and very slow to train. • Requires large training data sets. • Information from the beginning of a passage may be forgotten by the end.
Large-scale transformer networks	Similar to recurrent neural network but designed in such a way that all inputs can maintain many representations of all other inputs.	Context sensitive	High	• Larger contextual awareness • Best expressivity	• Large amounts of training data and computation resources required to train. • Success is very dependent on architecture and parameters.

Key Important Point

CAP data are mostly natural language; therefore, it is important to use AI models that are designed for text.

A key challenge with CAP data that separates them from common ML solutions in other applications is the heavy reliance on free-form text-based fields. Almost every important field in CAP data is a free-form text field, and any AI solution using these data will need to be good at dealing with CAP data. At a minimum, ML systems working with CAP data will need to transform the text data into a useful format (for example, bag of words or aggregated Word2Vec) for use in model training. The best performing systems will leverage newer state-of-the-art techniques in NLP that use large neural networks to capture more nuanced language information and interactions.

Key Important Point

Rules-based approaches struggle with text data.

An approach that is commonly proposed in lieu of ML is the development of rules-based systems. The proponents of these systems posit that they can create their own conditional logic that can route CAP data through to automate decisions. An example of this would be a system that looks for keywords x/y/z in a CR title and does some thing as a result. The problem is that although the expert may be confident that the rules are foolproof, experience shows that these rules consistently break down in practical applications with real-world data. A rule that looks for the presence of *high-pressure safety injection* as evidence of a nuclear safety impacting condition will often incorrectly flag many issues that have nothing to do with nuclear safety (for example, "Operator was late to the continuing training session covering changes to the high-pressure safety injection system operability determinations."). Rules-based systems sometimes attempt to combat these issues with ever more complex rules (for example, some keyword is present, but this other keyword is not present), but the complexity, inaccuracy, and cost of such systems quickly become barriers too large to overcome. Even simple ML systems, such as training a boosted decision tree model on text data transformed to bag of words, will outperform manually configured rules-based systems on almost all occasions.[7, 8, 9]

[7] Gerhman et al., "A Comparison of Rule-Based and Deep Learning Models for Patient Phenotyping," https://arxiv.org/ftp/arxiv/papers/1703/1703.08705.pdf.

[8] Hurriyetoglu et al., "COVCOR20 at WNUT-2020 Task 2: An Attempt to Combine Deep Learning and Expert rules," 2020, https://arxiv.org/pdf/2009.03191.pdf.

[9] Sebastiani, Fabrizio, "Machine Learning in Automated Text Categorization," 2001, https://arxiv.org/pdf/cs/0110053.pdf.

Section 3: Automating CAP

Training AI to automate CAP decisions is a complex multistep process that involves a mix of business process engineering, data integration, and ML.

What Is CAP Automation?

The scope of an ACAPS can vary but generally involves using AI to automate decisions within the CAP at an NPP. By automating these decisions, NPPs can reduce the manual effort required to meet CAP program requirements and improve process efficiency and agility.

The term *CAP automation* covers automation of several processes within CAP. The most common processes being automated today are described in the following sections of this report.

CR Screening

CR screening is a process performed on all newly generated CRs and involves the assessment of the documented issue or condition, as well as the assignment of various resulting tasks. CR screening will typically include the determination of whether a CR is a condition adverse to quality (CAQ), the group that will own the CR, and the generation of initial corrective actions including evaluations and corrective maintenance. CR screening is performed in a variety of ways across the industry, but common approaches include a central committee of plant personnel screening all CRs for an NPP, decentralized CAP coordinators independently evaluating the CRs in their respective domains, or a combination of these two approaches.

Maintenance Rule Functional Failure Screening

Maintenance Rule functional failure (MRFF) screening is a regulatory required screening performed by a licensee to determine whether a Maintenance Rule–scoped system, structure, or component can perform its intended Maintenance Rule function. The implementation of the Maintenance Rule program will vary between some licensees. Certain

implementations of the Maintenance Rule program include a regular review of CRs for potential MRFFs or CRs that can prevent a system, structure, or component from performing its defined Maintenance Rule function.

Trend Coding

Trend coding includes the application of various trend codes to CRs. These trend codes are often Institute of Nuclear Power Operations (INPO) or World Association of Nuclear Operators (WANO) performance objectives and criteria or site-specific trend codes. These codes are typically applied to aid in trending and help facilitate straightforward analysis of the heavily textual CAP data. Trend codes can be applied throughout the CAP process, but most are typically applied during initial screening of a CR.

Reportability Review

Reportability reviews involve reviewing a CR for potential regulatory impacts including potential regulatory reporting requirements. For NPPs under Nuclear Regulatory Commission (NRC) regulation, the covered reporting requirements may include event notification reports under 10CFR50.72, licensee event reports under 10CFR50.73, Part 21 reports, and other regulations. There is some variation between NPPs on how this process is handled, but it typically includes all or a portion of CRs being routed to a regulatory review group for additional screening and analysis. The entirety of the reportability review is unlikely to be handled through CAP automation, but it is possible for CAP automation to determine whether a CR should be routed for additional analysis.

Key Operating Experience

Development of an automated Maintenance Rule analyzer at Utility G.

At Utility G, the Maintenance Rule program is implemented by strategic engineers who perform a review of every CR with an equipment failure to detect potential MRFFs. Utility G determined the most appropriate initial solution to be a recommendation system that presented the model results on the existing MRFF review web application.

The MRFF determination process at Utility G had three potential outcomes: MRFF Yes, MRFF Indeterminate, and MRFF No. Based on this understanding of the MRFF process, a binary classifier was developed to predict either MRFF Indeterminate or MRFF No. The binary classifier model was trained on CR text and categorical features from both the CR and plant equipment associated with the CR. This model was developed as a neural network with the *PyTorch* library. Overall performance of this initial model would allow approximately 40% automation of MRFF determinations through the binary classification system. The output of this model was displayed as an informational label on the existing MRFF review web application to engineers to allow for a review and feedback period.

Applying AI to CAP Automation

Setting up an ACAPS is a multistep process that, at its core, involves training AI models to automate decisions. Although there are integrations, auditing, change management, and other complicating factors, the AI part of the system follows a relatively straightforward life cycle.

Gather Historical Data

The first step in training AI on CAP data is to gather historical data from your CAP system. The historical data maintained within your CAP system will be the source of all knowledge for the ACAPS. At most NPPs, there will be tens of thousands of CRs spanning multiple years, containing all of the inputs and outputs for decisions that an ACAPS will attempt to automate. Acquiring these data and making them available for training AI models is a prerequisite.

Identify Data Known at Decision Time

This step is critical to ensuring that an ACAPS functions correctly. The historical data gathered in the first step will have data that are entered throughout the CAP process, from initiation to screening to evaluation and eventually closeout. Although all of these data are available in the historical records, only a subset of this information will be available when the ACAPS needs to make its decisions. It is important to separate the fields that are static and available at decision time and those that are entered later because the AI models must be trained with only the fields that will be available to it.

Identify Data Reflecting Decisions

This step involves identifying which fields reflect the decisions being made that the ACAPS will automate. This is relatively straightforward; for example, if your ACAPS will automate the decision about whether a CR is a CAQ, the field reflecting this decision in historical data needs to be identified so that the AI model can train to predict it. In addition, these fields should generally not be used as inputs for any other decision in the ACAPS.

Train the ML Models

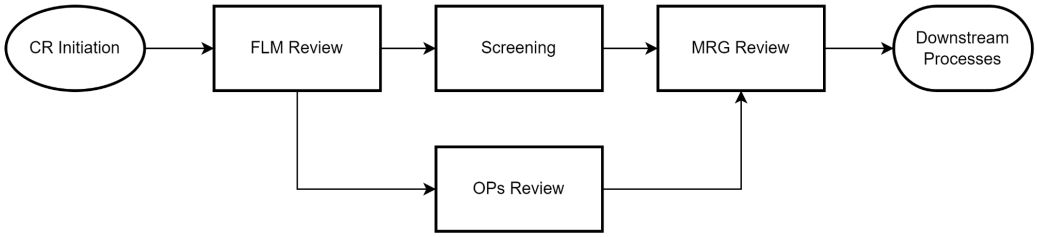
This step involves training the ML models that the ACAPS will use. The ML models are trained to use the available input fields from the historical CAP data to predict the decision outputs. How to train the ML models is generally the responsibility of a trained data scientist and is outside the scope of this report.

Integrate the ACAPS with the CAP System

This step involves integrating the CAP system into the ACAPS and its AI models. As new CRs are initiated, their information is sent to the ACAPS and fed through the ML models and the output decisions are sent back to the CAP system.

Inventory of Decisions in CAP

There are dozens of decisions across CAP that require manual effort and can be automated with an ACAPS. Figure 3-1 illustrates the screening process, and Table 3-1 presents a list of common decisions in CAP processes, the stage in a CAP process they are made, the estimated difficulty to automate, and the estimated value delivered by automating.



FLM = front line manager
MRG = management review group
OPs = operations department

Figure 3-1
Example CAP screening steps

Table 3-1
Typical decision fields encountered in a nuclear CAP

Decision	Description	Stage	Difficulty	Value
Equipment related	Typically a flag denoting whether the CR is related to equipment.	Initiate	Easy.	**Low.** Requires little time and expertise for manual review.
Procedure related	Typically a flag denoting whether the CR is related to procedure.	Initiate	Easy.	**Low.** Requires little time and expertise for manual review.
Operational impact	Either a flag or field denoting extent of operational impact.	Initiate/FLM review	**Medium.** High class imbalance and plant context needed.	**Low.** Requires little time and expertise for manual review.
Industrial safety related	Typically a flag denoting whether CR is industrial safety related.	Initiate	**Easy/medium.** Large number of examples available but need to have a good text model.	**Low.** Requires little time and expertise for manual review.
Condition severity/category	Field that indicates the severity of the condition and its effect on nuclear safety. Lowest level is usually non-CAP/not related to safety; higher levels indicate increasing levels of severity.	Screening	**Medium/high.** Class balance is better (usually 80/19/1), but determining severity requires understanding of equipment, impact, plant context, and many complex interrelated factors.	**High.** Usually the field requiring the most manual effort and discussion to determine; requires specific expertise to make determination.
Level of effort	Indicates the level of effort that the NPP plans to use to evaluate, resolve, and reduce risk of repeat events.	Screening/ management review	**Medium/high.** Level of effort is highly correlated with severity but sometimes differs for difficult-to-predict reasons (for example, this is a repeat issue and management wants to stop it from repeating).	**Medium.** Usually a consequence of the severity, but sometimes additional discussion goes into the *management discretion* part of determining level of effort.

Table 3-1 (continued)
Typical decision fields encountered in a nuclear CAP

Decision	Description	Stage	Difficulty	Value
Responsible group	Which group will be responsible for the issue, its evaluating, and resolution.	Screening	**Medium/high.** Training data are often inconsistent, and there is high output dimensionality.	**Low/medium.** Often, little time goes into determining the responsible group, but it is also often wrong. Line organizations are used to reassigning incorrectly assigned CRs.
Generated work activities	Which evaluations, action items, work orders, and so on will be generated to evaluate, resolve, and trend the issue.	Screening	**High.** The number, type, priority, and assignment of work activities all need to be predicted, and training data are highly inconsistent.	**Medium/high.** Much effort goes into generating work activities
MRFF	Determination of whether an issue resulted or could result in a system, structure, or component failing to perform its defined Maintenance Rule function. In some cases, additional justification for this determination can be provided.	Screening	**Medium/high.** Similar to condition severity. Extreme class imbalance can exist, and incorrect results are consequential.	**Low/high.** Proper determination of the MRFF can avoid engineering hours reviewing non-issues, but different plants devote different levels of effort to the process.
Regulatory impact	Determination of whether an issue will have a regulatory impact.	Screening/OPs review	**Medium/high.** Similar to condition severity. Extreme class imbalance can exist, and incorrect results are consequential.	**Medium.** Only higher severity CRs must be screened for this. High in cases in which plants may have a compliance resource screen all CRs for reporting requirements.
NRC/regulatory reportable event	Regulations indicate that certain types of events must be reported, including event notification reports under 10CFR50.72, licensee event reports under 10CFR50.73, and Part 21 reports.	Screening/OPs review	**Low/high.** NPPs have very few reportable events; therefore, training is difficult. Models tuned to minimal false negative rates have high success, however.	**Medium.** Only higher severity CRs must be screened for this. High in cases in which plants may have a compliance resource screen all CRs for reporting requirements.

Key Technical Information

Multilabel data sets are challenging but often needed in CAP automation.

One of the ML challenges in an ACAPS comes from the presence of multilabel classification tasks.

Multilabel classification tasks are AI problems in which a system must label a given example with between zero and many different labels. This is different from multiclass tasks in which the AI system chooses one label from a set of many, or binary classification tasks that discriminate between a record belonging to a single class or not. Multilabel classification tasks are particularly difficult because not only are there multiple possible labels for a given record, but a given record can also have more than one of these labels.

This challenge is often found in CAP automation, especially when applying trend codes to CRs. CRs often do not fall into a single bucket for trending and are given multiple trend code labels. An example of this is a repeat equipment issue identified during maintenance. This would likely have trend codes associated with equipment failures, maintenance, repeat issues, and even work management impacts.

Unfortunately, many ML algorithms struggle to work with multilabel data. Most historical research has focused on binary classification and sometimes multiclass classification. Fortunately, there are models (such as neural networks) that work very well with multilabel data sets and, when combined with newer architectures capable of state-of-the-art NLP task performance, can handle the multilabel task common in CAP automation tasks.

Review of ML Techniques for an ACAPS

Not all ML models and techniques will work well within an ACAPS, especially when considering the unique and challenging aspects of CAP automation. ML models that tend to perform well in this space fulfill many of the following criteria.

Criteria for Effective CAP ML Models

The following are criteria for effective CAP ML models:

- **Work well with text data.** CAP data are overwhelmingly text, and a model that struggles to handle text data will be unlikely to work well enough.
- **Support multiclass or multilabel output.** Most decisions in CAP automation are multiclass (more than one potential outcome) or multilabel (between zero and many different outcomes). Using models that do not natively support multiclass and multilabel outputs will result in many additional challenges in training, deployment, and auditing.
- **Produce confidence/probability of outputs.** Many of the auxiliary considerations for an ACAPS outside the core AI rely on having an interpretable confidence on the output decisions. Not all ML models do this well and instead produce only the most likely outcome. These will struggle to work well within the ACAPS.

- **Work well with cap data set sizes.** CAP data sets typically range from tens of thousands to hundreds of thousands of records. Some ML models may struggle with this amount of data, because it is either too large for them to deal with or too small.
- **Train in reasonable time and compute power.** CAP automation can result in substantial savings, anywhere from tens of thousands to hundreds of thousands of dollars (or more) every year. However, many partial implementations of CAP automation not tackling screening automation will be on the lower end of this range. As a result, models that would be very expensive to train or require large amounts of specialized hardware may not be optimal.
- **Inference time does not need to be optimized.** Unlike many use cases in AI, CAP automation models do not need to run in real time. There is often a significant delay between initiation, screening, and other downstream activities. Models that take seconds or even minutes to process a CR are acceptable, and there is little need for subsecond inference times.
- **Achieve high level of accuracy on cap automation tasks.** Even if all of the other criteria are met, an ML model that does not produce a high level of accuracy on CAP automation tasks will not be effective.

Assessment of ML Models

Table 3-2 provides an inventory of different ML models and anecdotal experience about how well these models meet the preceding criteria. This is provided as a useful starting point for developing or assessing CAP AI models.

Table 3-2
ML modeling approaches for nuclear CAP tasks

Modeling Approach	Fit to Preceding Criteria	Rationale
Decision tree	Poor	Decision trees struggle with multiclass and multilabel output and generally have lower accuracy.
Random forest	Poor	Although generally more accurate than decision trees, random forest models require transforming text data into either bag of words or a vector representation. Bag-of-words models struggle with accuracy, and although vector representation adds accuracy, the models still struggle with interpretable probability/confidence outputs and are not as effective on multiclass and multilabel problems.
K-nearest neighbor (KNN)	Poor to acceptable	KNN algorithms have very poor performance when dealing with text data because performance decreases exponentially as input dimensionality increases, and most text transformations for use in ML models result in very highly dimensional inputs. However, if custom similarity algorithms are developed (for example, weighted Jaccard cosine similarity), performance can be adequate, and multiclass and multilabel problems work well in KNN.
Boosted tree	Acceptable	Boosted tree algorithms such as XGBoost and CatBoost, perform reasonably well with mixed textual and nontextual inputs, multilabel problems, and small- to medium-sized data sets. Deep neural networks can often perform better if they are built correctly, but boosted trees can deliver acceptable results without as much modeling effort in many cases.
Naive Bayes	Poor to acceptable	Naive Bayes models make very strong assumptions about input data, mainly that all input data are assumed to be completely independent. This can reduce model performance and, importantly for automation, results in confidence predictions that are challenging to interpret. That said, Naive Bayes can perform adequately on certain text classification tasks, especially if the input data are prepared properly. In addition, Naive Bayes models work well on small- to medium-sized data sets and are not computationally intensive to train.
Support vector machines (SVMs)	Poor	Although SVMs can perform well when there is high dimensionality in the inputs, SVMs struggle with other aspects common in CAP automation tasks. SVMs can take a very long time to train when using tens of thousands of records or more as typical with CAP data. SVMs do not natively support multiclass or multilabel problems, and a model needs to be trained for each label. In addition, SVMs struggle when the data are not clearly separable (for example, there is no noise in the target labels). CAP data often have many inaccuracies and inconsistencies, especially near decision boundaries, that SVMs struggle to deal with.

Table 3-2 (continued)
ML modeling approaches for nuclear CAP tasks

Modeling Approach	Fit to Preceding Criteria	Rationale
Logistic regression	Poor	Although logistic regression meets some of the criteria for CAP automation, it struggles in many areas. Logistic regression often performs poorly with high-dimensionality inputs (for example, text) and does not natively support multilabel and multiclass problems. Critically, logistic regression usually delivers the worst accuracy compared to any of the other models covered here.
Deep neural network	Poor to best	Deep neural networks are the current state of the art for NLP classification tasks and can handle multilabel data seamlessly. However, deep neural networks often require specific expertise to train and deploy correctly and take significantly more computational power—and often specialized hardware—to train. Although they will deliver the best results when done correctly, it is quite possible to end up spending much effort with poor results.

Key Operating Experience

Ensemble modeling in Utility A's implementation of CAP automation.

For the development of its MRFF review and CAP screening automations, Utility A used a technique called *ensemble modeling* to achieve its results. Ensemble modeling is a technique in which multiple ML models are built and their outputs are combined into a single predictive pipeline. Utility A's use of this technique involved including the outputs of a Bayesian text confidence model and an artificial neural network in an ensemble multi-metric classification model (see Figure 3-2).

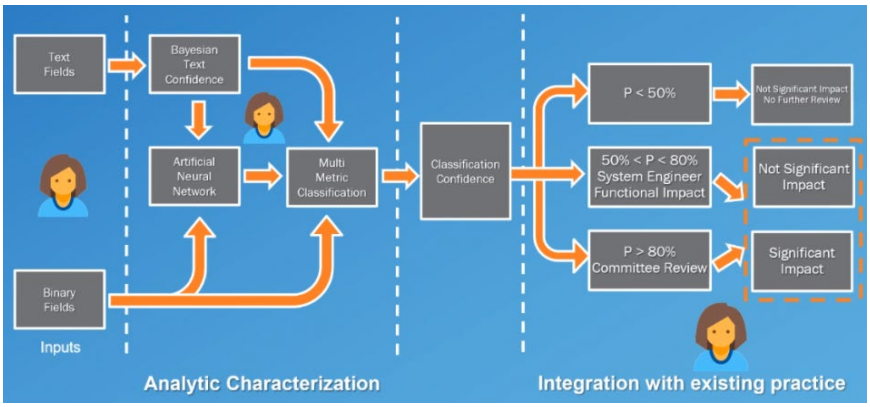


Figure 3-2
Applied ensemble modeling technique[10]

Key Cost/Value Considerations

Estimated costs of acquiring or developing task-specific models

A typical ACAPS will involve five-plus individual ML models for the different decisions, which results in a minimum estimated cost of $300,000–$450,000 for the ML models themselves over the first five years.[11]

[10] NRC AI/ML Workshop, https://www.nrc.gov/docs/ML2127/ML21277A139.pdf.

[11] https://www.phdata.io/blog/what-is-the-cost-to-deploy-and-maintain-a-machine-learning-model/.

Table 3-3 breaks down the cost of training a binary classifier for use in Maintenance Rule classification as performed by an experienced data scientist or ML engineer. Hour values are based on anecdotal experience.

Table 3-3
Estimated MRFF binary classification model costs

Steps	Hours	Line Item Cost @ $125/hour
Requirements gathering	20	$2,500
Data gathering and research	80	$10,000
Initial model development	100	$12,500
Test different modeling methods	120	$15,000
Model selection and refinement	50	$6,250
Total	***370***	***$46,250***

Section 4: Additional Considerations for an Effective ACAPS

In addition to developing reasonably accurate AI capabilities, there are several other key considerations to be made when designing and implementing an ACAPS. Although these considerations do not directly involve improving the raw performance of the ACAPS AI, they make a significant impact on the success of an ACAPS.

Business Impacts of Inaccuracy

An ACAPS delivers value by automating manual steps within the CAP process. The system has two types of additional costs beyond the cost of the system itself. Those costs are inaccurately automating a decision (false positive), which could cause an automation to occur incorrectly, and not automating a correctly predicted decision (false negative), which would cause the process to be performed manually.

False positives result in a cost to the business in the form of having portions of the CAP automated incorrectly. This cost is often difficult to calculate but is generally an estimate of the incremental increase in the cost of the factors listed next, as the result of a single CR being inaccurate. There are multiple components of this cost, including the following:

- Cost to reclassify later in process (if identified)
- Cost of spending more effort than needed to evaluate and resolve issues
- Costs from not spending appropriate effort to evaluate and resolve an issue (repeat events, equipment degradation/failure, latent human performance issues)

Some of these costs compound when the number of false positives increases and, in extreme cases, can include the following:

- Reduction in an organization's trust of the CAP and ACAPS
- Increased risk of INPO area for improvement
- Reduction in regulatory margin

In virtually all cases, ACAPS should be tuned to produce a low enough false positive rate that these extreme compounded costs are not introduced. As such, the initial direct cost of a false positive (for example, reclassification costs and incorrect resource allocation) is usually used for estimating the costs of an ACAPS.

False negatives, on the other hand, are not actually an additional cost themselves. False negatives result in a lost opportunity cost by not capturing the value of a true positive (a correctly automated CR). The opportunity cost of a false negative is the cost to manually process the decision. If a CR is manually processed and could have been automated, that is a cost that would have been avoided if the CR were automated. On top of this direct opportunity cost, the expected value of the cost of inaccuracy within the manual process should also be included. A CR that could have been automated correctly has the potential to be processed incorrectly in the manual process, and there is a cost to this risk of incorrect processing.

When evaluating costs, it is critical to recognize that these costs also exist when a record is not automated and existed before the automation system was put in place. The manual human processes used before CAP automation occasionally screened and processed CRs incorrectly.

Key Identified Best Practices

Compare ACAPS performance to a human level benchmark.

When adopting an ACAPS, stakeholders are often tempted to evaluate the accuracy of the system in a vacuum. When doing so, it is common to hear statements such as "we cannot automate any CRs if there is any risk of getting a single CR incorrect." These are unrealistic expectations and, unless tempered, will prevent any automation system from being used. That is why it is critical to not evaluate the ACAPS accuracy in a vacuum but rather to compare it to the current manual human-makes-the-decisions system. Based on anecdotal experience, the manually performed processes are 95–99% accurate on most decisions. This is the benchmark to which an ACAPS should be compared. If it is as accurate as the manual process, even if it is sometimes inaccurate, it can be comparable to current process quality.

Confidence and the Accuracy/Automation Trade-Off

In general, increasing the proportion of records automated will result in a decrease in the accuracy of those records, and decreasing the proportion of records automated will result in an increase in the accuracy (assuming that the automation model performance remains constant). This is because an ACAPS will first automate the most confident records that have the highest accuracy. Because only some records will be highly confident, to increase the proportion of automated records, less confident records will need to be included in the automation. These less confident records are less likely to be correct and drag down total system accuracy.

The curve adjacent in Figure 4-1 is typical of a classifier used in an ACAPS. As the green line moves to the right and all records are accounted for, the accuracy and confidence of the prediction determination declines. In this case, about 40% of the records would be automated with high confidence, 60% with at least medium confidence, and 40% with low confidence or not at all.[12]

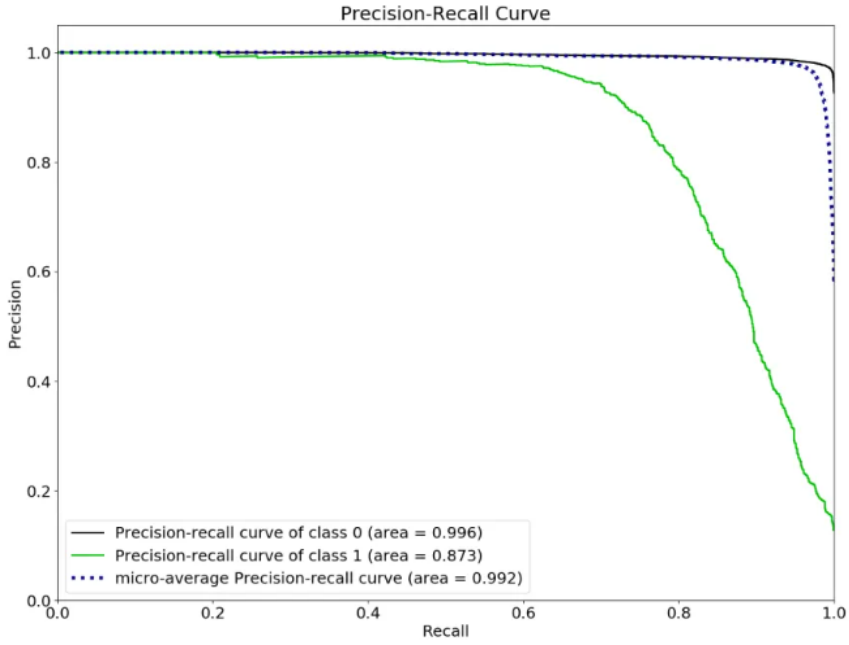


Figure 4-1
Example precision versus recall curve

From a pure cost optimization economics perspective, the optimal cost savings delivered through CAP automation would be at the point at which the marginal decrease in true positive value (additional value delivered through automation) associated with a decrease in confidence and accuracy matches the increase in false positive cost (cost of inaccuracy).

If the confidence probabilities of the system are calibrated appropriately, this point occurs at a confidence probability of the following (see Equation 4-1):

$$\frac{cost\ of\ false\ positive}{value\ of\ true\ positive + cost\ of\ false\ positive} \qquad \textit{Eq. 4-1}$$

[12] Czakon, Jakob, "F1 Score vs ROC AUC vs Accuracy vs PR AUC: Which Evaluation Metric Should You Choose?", https://neptune.ai/blog/f1-score-accuracy-roc-auc-pr-auc.

As an example, it can be assumed that the value of automating a CR (value of true positive) is $100 and the cost of an incorrect automation (cost of false positive) is $2400. In this case, the cost-optimal confidence threshold would be 96% (see Equation 4-2).

$$\frac{2400}{2400+100} = 96\% \qquad \textit{Eq. 4-2}$$

If 100 additional CRs were automated by changing the threshold from a higher value—for example, 96.1% to 96.0%—and the accuracy of those automations was 96%, there would be 96 automations and four inaccuracies, resulting in a value of $9600 (see Equation 4-3) and a cost of $9600 (see Equation 4-4).

$$Value = \$100 * 96 = \$9600 \qquad \textit{Eq. 4-3}$$

$$Cost = \$2400 * 4 = \$9600 \qquad \textit{Eq. 4-4}$$

This would be the exact point at which the accuracy/automation trade-off crosses. Any threshold above 96% leaves money on the table, and any threshold below 96% costs more in inaccuracies than in the automation value delivered. For example, if the threshold were moved from 96% to 95%, and as a result automated 100 more CRs, the incremental value would be $9500 (see Equation 4-5), but the incremental cost would be $12,000 (see Equation 4-6).

$$Value = \$100 * 95 = \$9500 \qquad \textit{Eq. 4-5}$$

$$Cost = \$2400 * 5 = \$12{,}000 \qquad \textit{Eq. 4-6}$$

This change would then have a net cost of $2500 (see Equation 4-7) and would reduce the overall value of the ACAPS.

$$MarginalValue = \Delta Value - \Delta Cost = \$9500 - \$12{,}000 = -\$2500 \qquad \textit{Eq. 4-7}$$

Key Important Point

The more decisions, the higher the likelihood of at least one error.

A challenge in CAP automation is that as more decisions are added to the processing of a single CR, the proportion of CRs that have at least one incorrect decision increases.

For example, imagine an ACAPS in which each decision can be automated with 95% accuracy. If the system automatically fills out three fields, the percentage of CRs with no errors is ~86% (95% ^ 3). However, if that system automatically fills out 10 fields, the percentage of CRs with no errors decreases all the way to approximately 60%. The number of correct predictions to obtain seemingly high levels of accuracy increases dramatically as additional fields are automated. This is especially true of fields that are hierarchical in nature, such as when predicting corrective actions, where there may be many corrective actions required—each of which can contain several additional fields with hundreds of independent choices.

Conservative Bias and Error Mitigation

Although the previous section outlines the cost optimal automation threshold, it would not be recommended to start an ACAPS using this automation threshold for several reasons. First, the previous optimization assumes that the confidence probability thresholds are perfectly accurate. Unfortunately, confidence probabilities are almost always miscalibrated because of differences between model training and real-world application. Second, a high-visibility inaccurate automation—even if truly offset by the delivered value—can be disastrous for stakeholder confidence and trust in the automation system and even cause additional regulatory scrutiny.

As a result, when adopting an ACAPS, it is generally advisable to maintain a conservative bias. Initial automations should start small in quantity by establishing very high confidence thresholds. As data are gathered and confidence is gained, automation levels should be slowly increased and manual sampling levels decreased. This helps mitigate the risk of errors and will still allow an NPP to reach near-cost-optimal levels, albeit on a slightly longer time period.

Key Technical Information

Watch out for model overfitting affecting classification probabilities.

In an automation system with confidence thresholds, it is critical that the ML models used produce useful and meaningful confidence values. Although many things can affect the produced confidence levels in a model, one of the key mistakes that can be made is overfitting. Put simply, overfitting occurs when a model memorizes its training data and therefore fails to predict unseen future examples accurately.[13] It is important to note that *overfitting* is usually used in the context of key model performance metrics such as accuracy or area under receiver operator characteristic curve, but in this context the reference is to the confidence values themselves—which is a more challenging metric to calculate.

It has been observed that if a model is parameterized to maximize the accuracy it will still overfit on predicted confidence, or *de-calibrate*.[14] In practice, this means that confidence values for predictions are pushed toward 100% or 0%. This push toward confident discrimination among target classes, although often improving overall accuracy, will increase the population of highly confident but incorrect predictions. Because an ACAPS relies on this type of error being minimized, overfitting predicted confidence can have significant consequences when automation systems are enabled. Therefore, it is important to reduce confidence overfitting by under-parameterizing models, under-training models that use batch learning, or directly measuring model performance against a customized automation loss function.

[13] https://en.wikipedia.org/wiki/Overfitting.

[14] https://en.wikipedia.org/wiki/Calibration_(statistics).

Section 5: Software Systems Integration

Although the discussion to this point has focused on the AI capabilities required for an ACAPS, the majority of the value of such a system can be achieved only through robust integration with the CAP management system.

Key Capabilities of an ACAPS Integration

A successful system integration between the CAP source system and ACAPS is critical. At minimum, it should include the following components:

- **Training data export/integration**. Allows the ACAPS to pull training data from the source system of record so that automation models can be retrained on a predefined interval or trigger.
- **Real-time or batch integration**. Allows the source system of record to pull automation predictions from the ACAPS or for the ACAPS to push new automation predictions to the source system of record.

If real-time integration is used, the following steps in the CAP workflow will need to trigger an action with the ACAPS:

- **CR initiation**. Enterprise asset management (EAM) system should push training data to the ACAPS and pull a prediction back into the CAP workflow if the FLM review is skipped.
- **Post-FLM review/pre-screening committee review**. Perform the same action as CR initiation if not already performed.
- **Post-screening committee review**. For records routed to a manual screening, push the human-applied values back to the ACAPS.
- **CR closeout**. Push any changes in applied values back to the ACAPS or provide a flag to the ACAPS indicating that target data have been updated.

Tracking of automations. Automation records and decisions should be tracked in a CAP source system, automation system, or both.

Feedback from EAM system. Feedback into automation system for auditing and additional training data.

Key Operating Experience

Integrating CAP automation with source systems at Utility G.

Although it is possible to adopt an ACAPS without integration into the CAP source system, many of the benefits are greatly reduced. When initially developing and adopting its performance objective and criteria labeling automation, Utility G did not start with an integration into the CAP source system. Instead, as the individuals performing trend coding were processing CRs, they would open a separate web application tied into the performance objective and criteria labeling AI models, input their CR info, get the predicted performance objective and criteria codes, and apply them manually in the CAP source system. Although this had the benefit of lower upfront costs and provided substantial flexibility, much of the efficiency savings from automation was negated by retaining the need to manually enter the predictions into the CAP source system. In addition, not all trend coders were as persistent about using the tool, and some decided to continue coding manually.

Integrated System Components

Enabling the key capabilities of an ACAPS integration will require interfacing with various existing IT systems. Integrations with EAM systems containing CAP data—such as Maximo, SAP, or Passport—will be a must. In addition to these back-end systems, NPPs may need to interface with other systems creating or interacting with CAP data. These can include web applications or mobile applications for capturing CR data or third-party vendor CAP applications.

Systems Architecture and Design

Ideally, an ACAPS should aim to enable as many key integration capabilities with the minimal degree of integration cost and complexity. In general, the more systems that are integrated and the more complex those integrations are, the higher the cost and risk of issues. When feasible, the ACAPS should aim to integrate directly with the CAP source system and avoid additional integrations. This limits the integration complexity and helps ensure data integrity (see Figure 5-1).

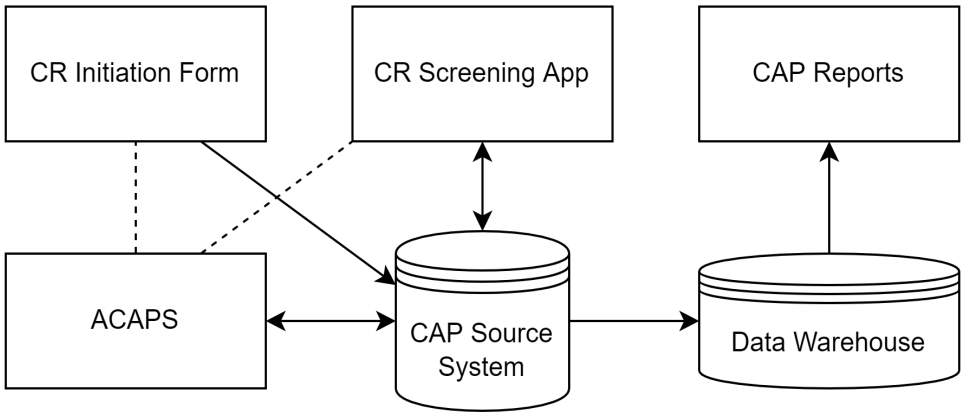


Figure 5-1
Reference CAP systems architecture diagram

In some cases, additional integrations beyond the CAP source system will be required—for example, a separate web application is used for CR screening, the automation confidence is needed to be displayed but the CAP source system does not support the addition of an automation confidence field. In this case, the CR screening web application would need to integrate with the ACAPS in some way to retrieve automation confidence.

Key Operating Experience

Using database procedures to integrate ACAPS at Utility G.

Utility G integrated its in-house ACAPS in two phases. The first phase exposed ACAPS recommendations to human screeners through a CR web application. The second phase integrated the ACAPS web application programming interfaces (APIs) directly with the EAM system. During recommendation, Utility G desired to fully expose the ACAPS outputs to the human screeners for review. The human screeners used a simple web application developed in .NET to review, screen, and update CR data. Utility G modified this application to include calling the ACAPS web APIs to retrieve automated CR output and added graphics to represent the output. When Utility G was ready to use ACAPS in a fully automated state, it desired that ACAPS be triggered as soon as a CR was entered into its EAM database. The Utility G's EAM is a custom system developed with Oracle Database. The ACAPS integration was added directly into the EAM database through database procedures and triggers executed at specific steps in the CR workflows. These procedures call the ACAPS through ACAPS web APIs, retrieve the information, and store it in the EAM. The CR workflow was modified to skip certain human steps that are now accomplished through ACAPS. The end result was that when a CR is entered, if it meets ACAPS criteria, it is automatically screened and advanced to the management review step with screening data applied.

Key Operating Experience

Integrating an IBM Watson–based ACAPS with site reporting at Utility D.

Utility D has developed its ACAPS in partnership with IBM Watson. Utility D has integrated ACAPS with its Asset Suite EAM and exposed the ACAPS results on the condition review group (CRG) report. Watson's APIs are called with CR data, and results delivered from Watson are inserted in a breakout section on the CR disposition report. The fields delivered are severity, priority, and owner group along with an explanation of when each field was selected.

Automation Error Handling

A critical part of an ACAPS is the ability to track and detect automation errors as a CR continues through the CAP process. An in-depth discussion of how error tracking is used from an auditing and monitoring perspective is covered later, but an important prerequisite for that auditing is that the errors are tracked and reported back as part of the system integration.

Errors in automated records from an ACAPS will almost always be reflected by a change to the automated CR. This can happen on any of the automated fields for a variety of reasons. Often, these are minor or inconsequential errors, such as the responsible group for a CR being changed because of available resources, a corrective maintenance work order being canceled because it is a duplicate of another, or the safety significance of a CR being changed after an evaluation reveals more information about the condition. Sometimes, however, these errors can be highly consequential, such as an audit finding that a CAQ was classified incorrectly and not evaluated.

These errors need to be considered and handled as part of the ACAPS integration. One approach for tracking these errors is to have an explicit step in the CAP workflow for changing consequential fields. Some CAP systems may already have this in place, such as sending a CR back to a screening step when safety significance needs to be changed. Although this approach is good at flagging high-consequence errors—and these errors can be tracked and sent back to the ACAPS through integration—it does miss smaller errors that may not have explicit steps in the CAP system. Another approach is to snapshot the CR at the time it is automated, either in the CAP system or in the ACAPs, and compare the final CR to this snapshot. This is an effective way to catch all errors in automation but may lack the ability to track the significance of the error or when the error was detected.

Front line organization tolerance for errors should be considered when designing the automation error workflow. Many organizations can self-reconcile minor errors such as incorrect responsible group assignment, especially if the group's responsibilities are similar or they are managed

under the same department. For these cases, some tolerance for errors is permissible and needs to be judged organization by organization; it may not even need to be explicitly tracked in the ACAPS integration.

ML Model Deployment Tools and Techniques

Deploying the ML models is a critical component of an ACAPS. An ACAPS must aim to meet the following requirements for it to be suitable for CAP automation:[15]

- Deployment of ML models in a robust server environment with API accessibility
- Ability to service up to a request per second during peak loads
- Ability to hot deploy ML models so that new models can be deployed with no downtime
- Ability to Alpha/Beta (A/B) test multiple ML models

ML model deployment software is still in its infancy, and the availability, applicability, and cost of various tools will likely change significantly over the coming years. Vendor systems that implement CAP automation are likely to provide their own embedded capabilities. Tools such as Seldon, Domino Data Lab, HPE Ezmeral ML Ops, and MLFlow are examples of software that may be useful for deploying ML models in an ACAPS.

Key Identified Best Practices

Few systems know how to deal with probability or uncertainty; care must be taken to handle this appropriately.

The CAP software systems that an ACAPS must integrate with do not inherently support the notion of confidence and uncertainty in the data they store. A CR is either a CAQ or not a CAQ or is an equipment reliability issue or not. An ML-based automation system, on the other hand, will rarely deal in absolutes. It is perfectly normal for an automation system to predict that "this CR has a 70% probability of being an equipment reliability Issue." This discrepancy can cause challenges in later processes, including potential human error traps as a result of misunderstanding the certainty of a field. At a minimum, records produced through the automation system should be flagged in the CAP source system. Storage and visual display of confidence for values provided by the automation system is an even better solution, albeit a more expensive option.

[15] Breck et al., "The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction," 2017, https://storage.googleapis.com/pub-tools-public-publication-data/pdf/aad9f93b86b7addfea4c419b9100c6cdd26cacea.pdf.

Key Cost/Value Considerations

Estimated costs of software systems integration, implementation, and maintenance.

The expected implementation cost for an ACAPS integration is between $5000 and $40,000, with anticipated annual maintenance costs in the range of 15–25% of implementation costs. CAP automation system integrations are generally cheaper than full application-to-application integrations because ACAPSs deal with one subset of data (CAP) and usually only have a couple of APIs that are called from specific parts of the workflow. It is impossible to provide a narrow estimate of the costs for a software systems integration because the integration costs will depend heavily on the system that it is being integrated with, the integration software being used, resource expertise, the ACAPS, and other intangible items.[16, 17, 18]

[16] https://www.starfishetl.com/blog/how-much-does-data-integration-cost.

[17] https://tray.io/blog/what-is-an-api-integration-for-non-technical-people.

[18] https://blog.dreamfactory.com/api-calculator-understanding-the-costs-behind-building-an-api-based-application/.

Section 6: Change Management

The change management for an ACAPS shares many similarities with any software project. An effective change management plan should aim to ensure that key stakeholders are aware of the upcoming changes, drive the desire to adopt the change, increase the knowledge of impacted parties concerning the affected processes, increase the workforce's skills with the new changes, and generally reinforce all aspects required to ensure that the changes are successful. However, change management for an ACAPS must deal with additional complexities related to automation and AI that are challenging to handle. It is recommended to use an incremental adoption framework in addition to other change management techniques to ensure the best chances of a successful ACAPS implementation.

Incremental Adoption Framework

An effective method for building confidence in an automation system is to use an incremental adoption framework. Using an incremental adoption framework allows owners of an ACAPS to clearly understand where they are in the maturity of their system and avoid costly missteps in adoption by moving too fast or missing key steps.

Although NPPs can adopt their own incremental adoption frameworks, it is suggested that they adopt the following framework consisting of five sequential steps: data, decisions, direction; assess; recommend; semi-automation; and automation:

1. **Data, decisions, direction** represents the initial stages of an automation system. This is the stage at which no system has been developed but stakeholders and subject matter experts have the opportunity to provide input on the automation system. As the name suggests, at this stage it is critical to ensure alignment across stakeholders on the following:
    a) **Data** that will be used to train the automation system and monitor its performance
    b) **Decisions** that will be automated through the automation system as well as any downstream impacts of those decisions
    c) **Direction** for the automation system and alignment on intended goals and appetite for accuracy/automation trade-offs

2. **Assess** represents the next stage of an automation and involves testing the automation system and ML models in a highly iterative, incremental, offline manner. Models are built, results are assessed, and adjustments are made. This is an important stage in automation system development because it is easy to catch many potential issues before they have made a negative impact and while they are cheap and easy to fix. This phase is also a good opportunity for educating stakeholders about how the automation system works and recognize its strengths and weaknesses.
3. **Recommend** represents the next stage of automation and entails bringing the automation system online but not automating any records or making significant changes to processes. There are several different approaches to this phase, but in general they involve recommending or defaulting to-be-automated fields and decisions in the source system using the automation system but continuing to perform the manual process. This is highly beneficial for several reasons. First, it provides an easy, robust way to assess the accuracy of the resulting automation system by recording which fields or decisions are changed during the manual review process. Second, this stage allows another opportunity to catch potential errors in the automation system and get feedback from key subject matter experts. Lastly, this phase helps produce a high level of confidence in the automation system because stakeholders gain further familiarity with how the system will work when they are no longer performing the manual processes.
4. **Semi-automation** represents the next step in automation and involves bypassing the manual processes for a subset of records matching certain criteria. These criteria vary and can include a mix of model confidence, keyword or other field blacklists and randomly selected samples of records. Semi-automation is important because it allows the organization to take small steps with the automation system and avoid over-automating and moving too far down the automation/accuracy trade-off curve too quickly. Like the recommend phase, this phase is valuable for building stakeholder confidence.

   The semi-automation phase typically includes the introduction of changes to underlying CAP procedures. At this point, the process and workflow have fundamentally changed; where there used to be a review of every CAP item, there are now automated bypasses and new processes for auditing and monitoring automation quality. It is critical that the underlying CAP procedures be updated to reflect these changes.

   Automation represents the final step in an automation system and involves the implementation of the automation system to its full potential. Sometimes, as is often the case with trend-coding automation systems where inaccuracies have a low cost, this involves automating 100% of records and no longer having a manual process. For other systems, such as condition screening automation, there is still some manual review for low-confidence predictions and audit purposes.

Key Operating Experience

Utility G CAP automation.

When Utility G deployed its CAP screening automation, it followed a very similar incremental adoption framework. First, the in-house data science team made an inventory of the available data elements—the decisions made by the Utility G screening committee—and decided on the direction for target accuracy levels and modeling techniques. Next, the team built prototype automation models, scored historical CAP data, and shared with key stakeholders. With stakeholder buy-in, it then deployed the automation system in a recommendation mode. When CRs were sent to the screening committee for review, they would be fed through the automation system and pre-populated with the fields needed to perform a review. After about one year in recommendation mode, the system was finally put into partial automation mode, with an initial ~10% of CRs bypassing the screening committee altogether.

Key Identified Best Practices

Allow for manual rules to bypass automation.

It can be advantageous during adoption of an ACAPS to include custom automation bypass rules above and beyond the automation system confidence. These rules are meant to catch certain types of CRs that an NPP is reluctant to process automatically, such as CRs referencing safety culture or CRs authored by audit organizations. It is generally trivial to write these manual bypass rules as part of the ACAPS or as an add-on to the integration process.

Other Change Management Practices

Parallel to the ideas covered within the incremental adoption framework, it is recommended that additional common change management practices be used to increase the chances of a successful ACAPS implementation. Some of the more impactful practices include establishment of risk tolerance, engagement with end users, and iterative development cycles.

Establishment of risk tolerance represents a common understanding of the acceptable levels of risk to the impacted organization. This is a natural part of the incremental adoption framework and is expected to change as adoption matures. This should be a well-established and intentional discussion at each step in the framework so that proper goals can be set at each level.

Engagement with end users represents the continuous involvement of the impacted individuals and teams through the implementation life cycle. Engagement with end users helps increase adoption of the change and can increase the cumulative impact of an implementation by leveraging the ideas and experiences of these users. This will also ensure future ownership of the system and long-term front line support for the changes brought on by an ACAPS.

Iterative development cycles represents introducing reasonably small changes to the existing CAP processes. These development cycles will ensure the independent success of individual ACAPS components, which will drive the success of the full implementation. The incremental framework stage of these components should be considered prior to embarking on a change to introduce a new component.

Establishing Ownership of the ACAPS

A key part of effective change management is establishing ownership and responsibility over new functions required after implementation of a project. This is even more critical when performing change management for an ACAPS because there are many new functions that do not exist in the existing CAP processes.

An ACAPS will have many of the same stakeholders of a traditional CAP system. These include the initiators, the CAP or organizational effectiveness groups, condition screeners, and a few additional groups including the IT organization and a data science, data engineering, a systems development group—if the automation system was developed in house—or the CAP automation system vendor. Close collaboration between the owners will be critical to the long-term success of the system.

There are many areas of an ACAPS that require the designation of clear ownership, including the following:

- Defining overall requirements of an ACAPS
- Defining threshold and accuracy requirements for process automation portions
- Defining software technical requirements such as architecture, deployment, and monitoring
- Requesting updates or making updates to the system as needed to align with user requirements or changing plant environments—how these changes and updates get installed into the system
- Approval of new automation components, such as ML models, or automation rules entering a production environment

An effective tool for ensuring that clear ownership is established is the use of a responsibility assignment matrix, or *RACI*. A RACI matrix describes the participation by various roles in completing tasks or deliverables for a project or business process. RACI is an acronym derived from the four key responsibilities most typically used: responsible, accountable, consulted, and informed. Although RACI matrices will look different for different NPPs, an example RACI is provided in Table 6-1 as a potential starting point.

Table 6-1
Example responsibility assignment matrix (RACI format)

Tasks	Plant Users	Human Screeners	CAP Team	IT Implementers	ACAPS Developers
Automated decision modification	Inform	Consult	Accountable	Responsible	Responsible
Identify changes to incoming data streams	-	Consult	Responsible	Consult	Accountable
Automation threshold modification	-	Consult	Responsible	Inform	Consult
Automation model retraining	-	Consult	Accountable	Inform	Responsible
ACAPS start-stop	Inform	Consult	Accountable	Responsible	Consult
Automation step phase change	Inform	Consult	Accountable	Responsible	Consult
ACAPS update	Inform	Inform	Consult	Responsible	Accountable

Section 7: Operating, Monitoring, and Auditing

An ACAPS is not complete without the implementation of the appropriate monitoring and auditing capabilities. CAP is a regulated process and automating portions of CAP requires that monitoring and auditing are in place to ensure performance and verify compliance.

Operation of an installed ACAPS will be similar to the current CAP screening process but with less manual time and effort required (see Figure 7-1). Exact operational details will depend on the current state of the implementation and the site requirements. ACAPS in a recommendation phase will be almost identical to the manual screening process, with few additional steps required.

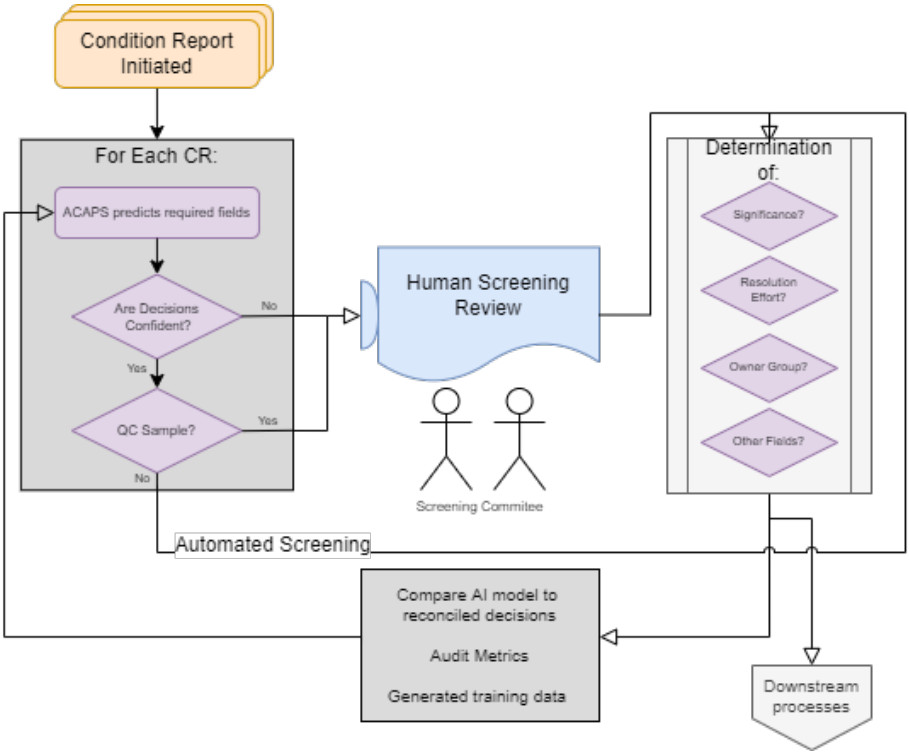


Figure 7-1
ACAPS basic operational flow

CRs will still need to be manually processed for a variety of reasons, as follows, to support an ACAPS in an automation or partial automation phase:

- **Not automated due to ACAPS performance**. In a partial automation phase, the ACAPS may not be confident enough to automatically process all CRs. Manual processing still needs to occur for these CRs.
- **Not automated due to data drift**. As an ACAPS continues to operate, it is likely that CRs will be produced that do not have any matching relevant historical data indicating how to process them. These will still need to be manually processed. More information about the impacts of data drift can be found in the Detecting Data Distribution Changes section.
- **Highly critical CRs**. NPPs will likely still want to manually process highly critical CRs, such as significant conditions adverse to quality (SCAQs). The CRs are typically complex and high impact, have significant regulatory and operational impacts, and are unlikely to have representative historical data. These CRs are so important that some NPPs will elect not to reduce total manual effort involved in CAP processes but instead reallocate saved time from ACAPS adoption toward discussing and evaluating these issues.
- **Later reviews and audit findings**. Various downstream processes, such as management review group reviews, CR evaluations, or even audits, may identify incorrect and controversial automated decisions. These records should be distinctly identified in the CAP source system so that they can be manually evaluated further. Reconciliation of these decisions helps collect future ground truth[19] information for additional ACAPS model training. This process is specific to the ACAPS implementation but can consist of later data ingestion through query, spreadsheet upload, manual flagging within ACAPS, or similar. This should occur only after all human review steps are completed and may or may not be accomplished in an automated fashion.

Although ACAPS can reduce the human effort required to support CAP processes, NPPs must be ready to support some level of continued manual operations. CAP processes, supporting software, and employees with the required skill sets to review CRs will need to be maintained, albeit in a diminished capacity.

Regulatory Impacts and Current Trends

When the first ACAPSs were put in between 2017 and 2019, there was minimal regulatory scrutiny. At the time, this technology was brand-new to the industry, and, even for those within the regulatory agencies familiar with the technology, there was little movement to monitor and regulate.

[19] *Ground truth* references the true value of an item as determined by a human expert.

Within the past year (2021), this has changed. The NRC is becoming increasingly aware of, and involved in, the use of AI and ML in the nuclear industry. In general, the NRC has been welcoming of the technology, acknowledging its value to the industry and its importance in improving safety and reducing costs. However, the NRC has indicated that it plans to draft a regulatory framework concerning the use of ML and AI within the industry in the coming years. Initial indications are that it will be focused on operations and direct plant-impacting technologies instead of administrative processes such as CAP. However, it will be important for utilities adopting CAP automation solutions to stay abreast of these developments and adopt best practices now to avoid potential regulatory compliance challenges in the future.

Section 8: ACAPS Quality Assurance

ACAPS Key Performance Indicators

There are several key performance indicators that are useful for tracking the performance of an ACAPS (see Table 8-1). Ensuring that these metrics are tracked and reported is critical to monitoring the health of the system.

Table 8-1
ACAPS key performance indicators

Key Performance Indicators	Description
Count of CRs automated	Tracks the number of CRs that are automated through the ACAPS.
Count of CRs manually sampled	Tracks the number of CRs that could have been automated but were fed through manual processes for quality control purposes.
Manual sample rate	$\frac{\#CRs\ manually\ sampled}{\#CRs\ manually\ sampled + \#CRs\ automated}$ Tracks overall sample rate. A high manual sample rate provides high levels of quality control and confidence in automation accuracy numbers but at the cost of not automating large numbers of CRs. Manual sample rate should be higher for young systems or systems that just went through a large change and should be decreased as system performance proves itself.
Automation efficiency	$\frac{\#CRs\ automated}{\#Processd\ CRs}$ This metric provides a useful gauge of how much work the automation system is automating.
Automation accuracy	$\frac{\#QC\ sampled\ CRs\ correct}{\#QC\ sampled\ CRs}$ This metric helps determine the accuracy of the automation system. It is typical for an automation system to aim to achieve the maximum automation efficiency for a minimum automation accuracy.

Key Technical Information

Stochastic AI models and model retraining present challenges with traditional software quality assurance.

Ensuring quality in an ACAPS has unique challenges. Although much of an AI-based automation system is built in software and can be tested through traditional software quality assurance processes, the key part of the automation system (the AI) cannot be tested this way. Testing that the automation system works correctly on several predetermined records does not guarantee that it will work on future data. Various approaches can help mitigate this risk, such as rigorous train/test/validate data splitting during model training and validation as well as simulating automations by holding out the latest time period of CRs—but these are still weak assurances of future performance.

As a result, it is recommended to add quality control–based testing post-implementation that randomly samples a portion of records that would have been automated and feeding them through a traditional manual process. The results of the manual process can then be compared to the predicted results, and the efficacy of the automation system can be tracked and monitored well after it is initially developed.

Audit Information to Track

It is critical that the automation system keep an accurate record of the threshold rules, active models, and overall automation configuration over time. To be able to support auditing, training record weighting, automation accuracy, and so on, the system must be able to reconstruct what the automation system configuration was at the point a CR was fed through the automation system. A well-designed implementation of an ACAPS would serve as the single source of truth for its configuration, including thresholds and automation rules, at any given time that an automated screening was performed. Recording of these details allows the ACAPS to serve as its own source of documentation and historical record at any time. If a system does not have the capability to record this type of information, the changes to thresholds and other key decision points should be included in a change document. The change document can differ based on NPP but should otherwise capture what threshold or other element is being changed, why it is being changed, and what the resulting effect is going to be. In the worst case, the minimum standard that should be enforced is a record of the control logic changes through system source code management. This allows historical information to be reconstructed, albeit at a high level of effort and time. A summary of points to be recorded each time a CR is automated is as follows:

- **Decision threshold value**. The probability/confidence level above or below which a model determines a specific classification.
- **ML model snapshot**. A reference to a model artifact that would allow exact recreation of the ML model that determined a decision.
- **Class labels**. The set of categories possible to be applied.

- **System configuration**. Reference to a summary record that documents the ML model versions in use at a given time. This record ties all other records together under a common reference so that the exact state of the system when an automated decision was made can be recreated.
- **Feature data record**. A copy or a reference to the feature data that were used.

Key Important Point

Every record automated is now a record the ACAPS does not learn from.

Although sampling of automated records is important for quality control and model performance, there is another key reason for sampling: gathering new training data. After a subset of CRs become automated and decisions are no longer manually made, no new training data are being generated. Automated records must be excluded from future model training because they would cause a positive feedback loop in which the models become incorrectly more confident in the predictions by being shown training examples that are actually just regurgitating information the model already knows.

It is important to manually sample some automated predictions to check model performance and continue to generate new training data. The amount of manually sampled data should be user configurable. Sampled records should be evaluated through the standard screening process in a single-blind format so that the reviewer does not know that he or she is reviewing a record marked for quality sampling.

By sampling some portion of the automated records, even if not needed for quality control purposes, training data continue to be generated for all subsets of CAP data (although at a lower rate for the automated subsets). Because the underlying processes and decisions driving CAP automation are effectively guaranteed to change over a long enough time period, these training data are crucial. Eventually, model performance will degrade, resulting in the CAP automation models needing to be retrained. At that time, it will be critical to have at least some new data to train the models.

Section 9: AI System Monitoring

Model Performance Measurement and Tracking

Model Key Performance Indicators

Key performance indicators that should be used to measure performance for automation-based models will differ from many other applications of ML. Because the key metrics being optimized are automation efficiency and automation accuracy, it is important to select and monitor metrics that track these results. Measures such as raw accuracy are unusable here. Between class imbalances and partially confident predictions, it is easy to have a highly accurate model that is a poor model for automation.

Although far from an exhaustive list, the following metrics should provide sufficient information for evaluating the performance of ML models for CAP automation:

- **True positive**. A correct prediction of the positive class.
- **True negative**. A correct prediction of the absence of the positive class.
- **Precision**. Precision is perhaps the most important metric for evaluating model performance. Precision is defined as shown in Equation 9-1:

$$\frac{\#TruePositives}{\#TruePositives + \#False\ Positives} \qquad \textit{Eq. 9-1}$$

  In the context of an ACAPS, this metric will be an estimate of the accuracy of the model for records that are automated if the same confidence threshold is used for the decision boundary.

- **Recall @ minimum precision.** Recall is the percentage of all positive records that are true positives (see Equation 9-2):

$$\frac{\#TruePositives}{\#TruePositives + \#FalseNegatives} \qquad \textit{Eq. 9-2}$$

  Recall has an inverse relationship with precision; so, it is useful in the context of ACAPS to evaluate the recall metric at the minimum precision (*minimum precision* being the minimum acceptable automation accuracy). This metric provides an estimate of the percentage of records that will be automated as well as the efficiency gains on the system. Figure 9-1 highlights how a selected precision

value (labeled *classification threshold*) affects the recall potential of the system. As the threshold moves left, more green records are identified by the model at the cost of additional red records. Setting the classification threshold to a minimum precision value allows a user to balance the amount of automation versus the accuracy of the ACAPS.

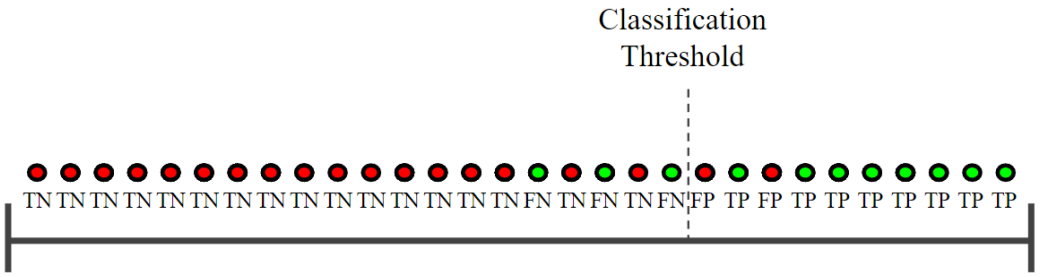


Figure 9-1
Selected precision value on a number line[20]

- **F1 score**. The F1 score is useful for evaluating model performance for models in which the importance of true negatives is minimal and is calculated as shown in Equation 9-3:

$$\frac{\#TruePositives}{\#TruePositives + {}^{1}/_{2}(\#FalsePositives + \#FalseNegatives)} \qquad \textit{Eq. 9-3}$$

  CR trend coding is a common example of this because there are sometimes hundreds of codes not applied for each code that is applied—and metrics taking credit for true negatives vastly overestimate model performance.

- **Area under curve (AUC)**. A common measure of a model's ability to deliver accuracy as well as separate high-confidence predictions from lower confidence predictions. The AUC calculation is complex because it involves ordering records by their output confidence, calculating the running precision across all records, and then calculating the area under the running precision curve. AUC is not directly interpretable, but it is a great metric for comparing different candidate models. Models with higher AUCs on test data will almost always perform better in an ACAPS.

Quality Control of Automated Records

For an ACAPS that requires the ability to monitor accuracy or performance, the recommended approach is to implement quality control measures that involve random sampling of automated records. This involves feeding data through the automation system, determining whether a record will be automated, and—of the records that will be

[20] Classification: Precision and Recall, https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall.

automated—choosing a random sample of these records to still be fed through the manual process. After they go through the manual process, the results can be compared to what the automation system would have done—and conclusions can be drawn about the accuracy of the automation system.

The strength of using this approach is that methods from statistical process control[21] and best practices can be applied. The full scope of quality control sampling plans is outside the scope of this report, but the implementation of any common sampling plan will allow the organization to set a minimum acceptable quality level for a given number of records—and a sampling rate can be provided to ensure that that level is met.

A weakness of the random manual sampling approach is that organizations often believe that its manual process is deterministic and that there are never errors in the manual process. In the majority of cases, this is demonstrably false because different individuals can and do make different disposition determinations for an identical CR. This can be the result of unclear decision boundaries, changing contexts, variations in training and experiences among individuals, or even mis-keying entries in the system. Anecdotal experience indicates that in CAP systems, the error rate can vary from 1% to 2% for highly important fields and all the way up to 25+% for trend codes. As a result, if the acceptable quality level is near or lower than the error rate of the manual process, one will struggle to achieve that level because it will be challenging to attribute errors to the manual process or the automated process.[22, 23, 24]

---

[21] https://en.wikipedia.org/wiki/Statistical_process_control.

[22]Ginart et al., "MLDemon: Deployment Monitoring for Machine Learning Systems," 2021, https://arxiv.org/pdf/2104.13621.pdf.

[23] Re et al., "Overton: A Data System for Monitoring and Improving Machine-Learned Products," 2019, https://arxiv.org/pdf/1909.05372.pdf.

[24]Klaise et al., "Monitoring and Explainability of Models in Production," 2020, https://arxiv.org/pdf/2007.06299.pdf.

Key Identified Best Practices

Choose a manual sample rate and precision threshold to bound the worst-case model degradation.

After systems have been designed to incorporate manual sampling and precision/confidence/probability thresholds, selection of these values in a conservative fashion will bound downside risk of an inaccurate model. A manual quality sample rate of 100% of automated records effectively puts the system in a recommendation-only model in which all records retain manual review. A manual quality sample rate of 0% automates all records and captures no information about model performance. Values in between will allow the ACAPS to balance monitoring of model performance with exploiting the automation efficiencies. A recommended long-term point would be somewhere between 5% and 10%.[25]

Detecting Data Distribution Changes

What Is Data Drift?

Data drift represents a change to the data and patterns for an ML model. Data drift is particularly dangerous in automation systems because the ML models are trained to detect specific patterns in data. If the data and patterns change in any meaningful way, automation accuracies are likely to suffer.

Data drift can happen in several different ways. Examples of common data drift patterns are as follows:

- **Concept drift**. The relationship between the input data and the output has changed. An example of this would be a change in the CAP procedures to now classify certain CRs as CAQs when historically these CRs did not used to be classified as CAQs.
- **Prediction drift**. The patterns in inputs and outputs remain the same, but the frequencies have changed. An example of this would be a large increase in CAQs related to primary systems within containment during an outage.
- **Feature drift**. Significant changes to input data are seen. An example of this might be seeing shorter condition descriptions after the adoption of a new mobile app for creating CRs.

[25] Rate selection may be chosen on a batch basis from acceptable quality limit charts in American National Standards Institute Z1.4 or through continuous statistical process control measures in which the mean incorrectness (defect) rate is tracked. Sampling rate schedules may also be determined. See sources:
Dodge, H. F., "A Sampling Plan for Continuous Production," 1943.
Bebbington et al., "Continuous Sampling Plans for Markov-Dependent Production Processes under Limited Inspection Capacity," 2013.

Data drift often occurs as a result of typical business activities, but errors in system integrations and changes to source systems can cause significant unplanned data drift. An example of this would be a change to the CR entry form logic that puts the condition summary into a new field and the interface with the ACAPS is not updated to reference the new field.

The danger of data drift is that the incoming CRs and their relationship to CAP decisions now look different from the data the AI models were trained on and, as a result, the patterns learned may no longer be correct. That can result in incorrect predictions and confidence levels being produced by the models. Even small changes in one input feature can have significant consequences as ML models track complex patterns of interdependencies between inputs. This is often referred to as the *CACE principle*: changing anything changes everything.[26]

Monitoring for Data Drift

Monitoring for data drift inside the ACAPS is an important capability for mitigating risk. Automated monitoring for data drift usually entails the use of statistical tests or additional ML models to detect changes in input data distribution.

Key Identified Best Practices

Mitigate data drift by identifying impacts at the source and preparing the models and system for change.

After an ACAPS is deployed, it is critical that the ACAPS owners be part of any configuration and change management for the CAP source system and CAP processes. This section outlines many strategies for monitoring for data drift, but many of these strategies require time and preparation and are significantly less effective when used reactively. An ACAPS with engaged owners staying abreast of changes to underlying systems and proactively controlling for data drift will have higher automation accuracies and fewer negative impacts.

Statistical tests for data drift involve tracking input data and performing statistical tests over various time periods to identify whether any statistically significant changes have occurred to the input data distribution. Generally, the statistics should compare new data with the data sets used for training the ACAPS models. A full inspection of data drift statistical tests is outside the scope of this report, but commonly used tests include ADWIN, Population Stability Index, Kullback-Leibler, Jenson-Shannon, and Kolmogorov-Smirnov. These statistical tests are fairly robust but track only significant changes across the entirety of data coming in. As a result, models will typically need to be retrained to resolve the data drift impacts.

---

[26] Sculley et al., "Hidden Technical Debt in Machine Learning Systems," https://wiki.esipfed.org/w/images/5/5f/NIPS-5656-hidden-technical-debt-in-machine-learning-systems.pdf.

ML techniques involve the use of anomaly detection algorithms and autoencoders and are generally applied on each incoming record. Autoencoders, KNNs, and other approaches are commonly used. Although the addition of more ML models to the ACAPS can increase complexity, the advantage of this approach is that data drift can be caught on a record-by-record basis. As a result, only the records outside the original training data distribution need to be manually screened, and the ACAPS can continue a slightly lower automation efficiency.

Key Identified Best Practices

Use anomaly detection to determine whether similar data have been seen before.

In general, ML systems struggle to perform well on data that are highly dissimilar to the data on which they were trained. Most ML algorithms are unable to indicate whether they have seen similar data before and may erroneously produce highly confident but incorrect predictions on these examples. Therefore, it is advisable to include an anomaly detection algorithm in the system that can prevent novel records from being automated.

Application System Monitoring

Automated monitoring of the ACAPS is recommended as the reliance upon the system increases. The application should be integrated with existing monitoring tools within the organization to provide automated alerting and performance issue resolution.

Non-AI-Related Key Performance Indicators

Monitoring of the systems enabling the ACAPS focuses on ensuring that the AI system remains available. Monitoring of these services may take on similar characteristics of monitoring other critical system services a utility may have enabled such as an EAM system or grid management system. Another example would be a critical web service that exists on the commercial market such as flight booking or commercial financial applications. These metrics are not aimed at AI performance or accuracy metrics but rather focused on the systems that enable the AI to function. Monitoring metrics for these systems can include things such as those shown in Table 9-1.

Table 9-1
ACAPS operational key performance indicators

Key Performance Indicator	Description
Throughput	Throughput measures the amount of traffic that is flowing through the automated AI system. Attention should be paid to times of increased demand such as a refueling outage or audit period.
Response time/latency	Response time measures how long the CR screening application takes to respond to requests. Lower response time is better, and the application should be optimized to the point that upstream or downstream systems do not time-out or are adversely affected.
Error rate	The amount of error responses delivered by the AI application back to the upstream system. Errors can occur due to malformed requests, data that are not formatted properly, or other reasons. These errors should be logged, quantified, and addressed.

Key Identified Best Practices

Build in emergency stops that can be activated in case of emergency or catastrophic failure of the ACAPS.

When deploying an ACAPS, it is advisable to implement an emergency stop. The intent of this emergency stop button is similar to one on an assembly line: immediately stop the automation of CAP processes to minimize impacts of an issue within the automation system.

Although unlikely, there are circumstances that could entail the use of the emergency stop functionality. A non-exhaustive list includes a retrained ML model noted as having worse/unexpected performance, changes to a field in the source system resulting in sudden degradation of automation accuracy in new records, or an error in automation logic causing certain records to be automated that should not be.
In each of these cases, an emergency stop that can be quickly activated allows an NPP to minimize the impact of one of these situations while a solution is developed.

System Availability Requirements

In the beginning stages of ACAPS adoption, system availability requirements are generally not stringent because the vast majority of CRs are still manually processed. In the event of an ACAPS outage, it is expected that CRs will simply default to the manual process.

However, after the ACAPS processes a higher percentage of records, the impact of an outage becomes significant—especially if it is an extended outage. For intermittent system availability impacts (for example, outages less than a day for maintenance), it is expected that CRs that would go

through the ACAPS would need to be queued up and reprocessed through the automation system when it is available. This assumes that queueing is acceptable, which is easy to imagine when comparing to today's as-is processes but can have a significant impact as more downstream processes rely on the real-time automation of CAP program tasks. As a result, it is important to develop an ACAPS that can support high availability requirements even if not leveraged initially. An example of this would be building a system that can be run on redundant servers in a high availability setup and a disaster recovery plan but potentially starting with only a single server to reduce costs.

Implementation Technologies for Monitoring and Logging

Monitoring and logging technologies may feed an existing enterprise alerting or notification system. For monitoring ML systems, the monitoring and logging technology should contain a live updating dashboard that contains displays of the relevant metrics for real-time or near-real-time monitoring and a feed into the existing alerting system.[27] A monitoring and logging technology stack could be broken down into the following three categories:

- **Developed in-house**. Tools implementing solutions for each of the detection mechanisms previously covered. This may be common to implement specific monitoring techniques that may not yet exist in open source or commercial ML monitoring solutions. These solutions would be deployed on their own or by a serving mechanism such as *Kubernetes* or *Native* or on a cloud system. A serverless style of deployment allows these to remain active and efficient.[28] Logs and metrics of these systems could be exposed through a tool such as Prometheus[29] and visualized with a dashboarding tool such as Grafana.[30]
- **Traditional software application monitoring tools.** These are the well-known, traditional services used to monitor enterprise applications that could also be used for ML monitoring. Examples of these are *Elastic, Logstash, Kibana* (ELK) stack, which can be used to collect and visualize created logs or commercial offerings from companies such as Datadog, Traceview, or AppDynamics.

---

[27] Breck et al., "The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction," https://storage.googleapis.com/pub-tools-public-publication-data/pdf/aad9f93b86b7addfea4c419b9100c6cdd26cacea.pdf.

[28] Klaise et al., "Monitoring and Explainability of Models in Production," 2020, https://arxiv.org/pdf/2007.06299.pdf.

[29] *Prometheus* is a an open-source monitoring system with a dimensional data model, flexible query language, efficient time series database, and modern alerting approach, https://prometheus.io.

[30] *Grafana* is the open source analytics and monitoring solution for every database, https://grafana.com.

- **Tools with an ML focus**. Tools developed specifically to monitor ML workloads. The following are projects or vendors with a specific offering:
  - **Tensorboard** visualizes model training and logging during development.
  - **Seldon.io** combines model metric logging and visualizations into a purpose-driven tool.
  - **Weights and Biases** allows model monitoring and experiment tracking.
  - **Arize** provides a model troubleshooting tool with various ways to visually slice model performance data for monitoring and troubleshooting model performance issues.

Estimated Costs of Monitoring and Logging

Key Cost/Value Considerations

Saving on monitoring and logging costs.

Although monitoring and logging are important, ACAPS project costs can be reduced by omitting any production monitoring and logging or by integrating with an existing enterprise monitoring and logging system. Choosing not to perform ACAPS monitoring and logging will result in increased risks to the project, but if properly mitigated they are not excessive. Risks can be mitigated by placing automated triggers that fall back to the human system or putting conservative modeling thresholds in place.

Monitoring and logging costs will heavily depend on the option that is selected. Existing application monitoring and alerting are highly likely to exist at an enterprise level and will not be covered. Standing up and maintaining an ELK stack specifically for monitoring an ACAPS will cost around $8500 per year after computer and human resource costs. Tools focused specifically on ML will trade general monitoring abilities for curated, specific model metric and performance monitoring and may cost significantly more. For example, *Seldon.io* costs about $18,700 per year for five monitored models. A fully managed solution such as *Fiddler* or *Arize* will have costs comparable to *Seldon.io*. Cost estimates for an open source and customized solution are provided in Table 9-2.

Table 9-2

Monitoring solutions costs

Monitoring Solution	Software Licenses per Year	Compute per Year	Labor
ELK[31]	$0	$2388	$6360
Seldon.io[32]	$15,600 for five models	$3179	$0

[31] https://calculator.aws/#/createCalculator/OpenSearchService Using c4 large instances without ultrawarm.

[32] https://aws.amazon.com/marketplace/pp/prodview-tnyp2h3acabm6?sr=0-1&ref_=beagle&applicationId=AWSMPContessa.

Section 10: Maintenance and Sustainability

After an ACAPS is established, efforts are needed to ensure that the application continues to function and remains a sustainable ongoing effort.

ACAPS Oversight and Governance

ACAPS Ownership After Implementation

A potential challenge after the deployment of an ACAPS is determining which groups in the enterprise take ownership.[33] Although project management and change management efforts will often identify ownership during initial implementation and go-live, ACAPS ownership requires clear ownership for long-term sustainability. Although an ACAPS has characteristics of traditional software, there are unique aspects of long-term maintenance that require special care and oversight.

With traditional software, the organizational unit (for example, CAP group, Org Effectiveness, and so on) will provide significant input into the initial requirements and testing of the software. After deployment, however, there is typically limited involvement from the organizational units at the NPP. Most software does not typically require constant monitoring, nor does it tend to require updates. Most changes to business processes that are supported by software are typically managed with minimal change to the software systems. These changes are performed with some change management, communication, and training but generally only for those parts of the organization directly affected. Critically, in the context of CAP automation, the business logic for determining effort levels, safety significance, and responsible groups exists completely outside the software and can be altered by process document changes.

With an ACAPS, logic and rules that used to be part of the business process and executed by humans are now part of the automation system. Processes to set and maintain this logic—such as thresholds, logical decisions, and rules—need to be determined, owned, and controlled by

[33] In this case, *ownership* means responsibility for the function of.

the organizational unit responsible for the CAP process. Clear ownership and responsibility for this process becomes more critical than ever, and most business process changes can no longer be made without corresponding changes and impacts to the ACAPS.

Although, typically, the ownership of software systems belongs within IT, the ownership of and the results produced by an ACAPS should be the sole responsibility of the CAP organizational unit. A well-designed ACAPS implementation will expose these software controls and enable the CAP program owners to make CAP process changes as needed without the involvement of an IT or data science group. Presuming that the ACAPS implementation provides proper insight into the effects of application changes on the overall process, allowing a CAP organization to maintain these controls most closely resembles the governance and control schemes of current human-driven processes.

Key Important Point

If you want to change how decisions are made, you no longer train people, you train the model.

It is important to understand that when the majority of your CAP process is automated, making changes to the CAP process must now be performed by changing both the people process and the automation process. The sustainability of the CAP program will depend on both changes working in sync. Changing how your CAP process works is no longer performed by changing your procedures and training individuals but rather by generating new training data and updating your ACAPS. Specific strategies for updating the ACAPS for CAP process changes are addressed later, but understanding this point is critical toward understanding how to maintain and sustain your ACAPS.

The remainder of the responsibilities required for implementing the ACAPS should be defined and assigned according to Table 6-1, Example responsibility assignment matrix (RACI format).

System Maintenance and Updates

Updating the ACAPS Models

An ACAPS will need to retrain or update periodically for a variety of reasons, including updating to handle data drift and increasing automation accuracy by learning from new data. A variety of strategies exists for updating AI models in production. The correct strategy will depend on the specifics of your ACAPS and NPP. The strategies are as follows:

- **Online training**. Models using online training continuously train as they see new records. This is commonly what people believe AI systems do, but, in practice, online training is infrequently used. The benefits of online training are relatively minor in the context of an ACAPS and can potentially result in minor increases in accuracy. The potential risks and additional complexity, however, are immense. Online training means

that your system is constantly evolving and assessing, and troubleshooting becomes significantly more challenging. In addition, many ML techniques do not support online training; so, even implementing it in the first place is a significant challenge.

- **Periodic training**. A commonly used method for retraining is to retrain on a periodic basis such as quarterly or annually. This approach is fairly robust and has a good blend of maintaining system performance without introducing undue complexity. This will guarantee that models are never too far behind any changes to underlying data distributions and decision processes. The risk with this approach is retraining too often and introducing unnecessary changes to the automation system or not retraining frequently enough to learn new changes.
- **Metric-based retraining**. Metric-based retraining involves setting various thresholds on system performance metrics and retraining when those thresholds are met. These thresholds could be accuracy based, record count based, data-drift based, or based on any number of other measurements. This approach depends heavily on the metrics and thresholds chosen. When done correctly, this can result in near-optimal system performance but if done incorrectly can result in training far too often or not nearly often enough. Some risk from this approach can be mitigated by combining with bounds on periodic training—for example, not training more often than once every two weeks and never going more than a year between retrainings.

Key Identified Best Practices

Exclude automated data from future training data to avoid positive feedback loops.

A positive feedback loop occurs when an ACAPS model is trained on records it had previously automated that are now considered ground truth. Training repeatedly on previously learned decisions can potentially reinforce incorrect decisions and reduce the diversity of previously learned decisions. Automated records should be notated so that future persons developing or working on the ACAPS understand that these records were generated using an automated process and therefore should be ignored or reviewed prior to being used in training a new model.

Positive feedback loops are considered technical debt in any ML system and should be avoided to prevent the current system state negatively influencing future behavior. Isolation of automated records from the ML training system serves as a method to eliminate negative effects on future model performance.[34]

[34] Sculley et al., "Hidden Technical Debt in Machine Learning Systems," https://wiki.esipfed.org/w/images/5/5f/NIPS-5656-hidden-technical-debt-in-machine-learning-systems.pdf.

Key Technical Information

Weight training data post-automation to calibrate model probabilities.

When retraining models that include periods in which the ACAPS was in effect, it is important to understand and control for the effects of automation and manual sampling (see Figure 10-1). Because records that were automated from the new training data must be excluded, the distribution of the training data is altered and the effect of this must be controlled. It is critical to control for this effect by weighting the training data, oversampling manually sampled records, or undersampling non-automated records.

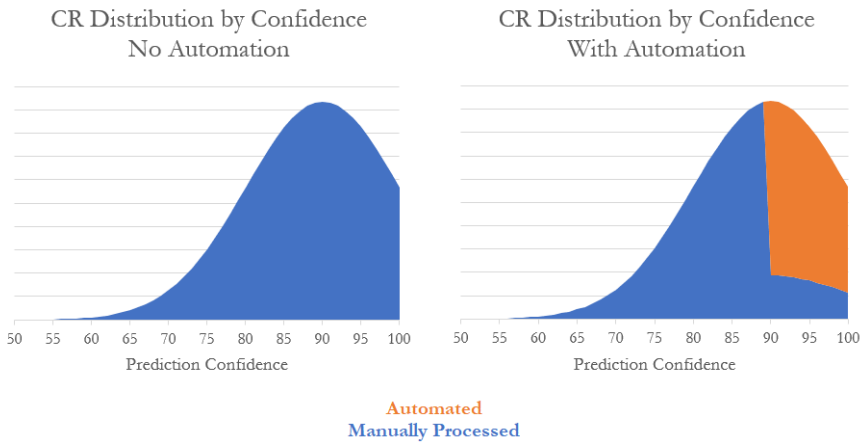


Figure 10-1
CR prediction confidence distribution changes due to automation deploying updated models

After a new model has been trained, there are a variety of strategies for deploying it. Deploying a new model presents an opportunity to improve the performance of the ACAPS and at the same time presents a risk that things will not go according to plan. The following deployment strategies attempt to maximize the value and minimize the risk of deploying an updated model into production:

- **Cutover**. Cutover describes the disabling of an existing model in production and a new model being immediately activated to replace it. Increased monitoring and a temporary increase in quality control sample rate are recommended to use a pure cutover transition.
- **A/B test**. An A/B test will run a new model and an old model in unison. The newer model is run using a smaller percentage of records, results are gathered, and, if the new model performs well, the percentage of records fed to the new model is increased and vice versa.
- **Reinforcement learning**. A more elegant version of A/B testing, a reinforcement-based approach allows you to introduce multiple models at the same time. A reinforcement learning system will explore how new models work by feeding some records through and tracking how well they do. As more evidence is collected, the

performance of specific models can be inferred. Models that perform well are used more often, eventually near 100% of the time. Models that perform very close to one another are chosen at random. Although more flexible and potentially more performant, a reinforcement-based system is much more complicated. If only one or two models are in consideration, it is likely easier to do regular A/B testing or even straight cutover with monitoring.

Key Operating Experience

Utility G and the multi-armed bandit.

During the deployment of an automated trend coding solution, Utility G needed to select a strategy for determining how many trend codes should be automatically applied to a CR. Using a simplified reinforcement learning method called the *multi-armed bandit*,[35] it was determined how many trend codes output by a single model would be used by the automated trend coding framework. The model began by selecting randomly from one of five result selection strategies and over a period of two weeks determined that the user's preference was a simple Top N selection strategy where *N* is the number of applied codes.

Adverse Change Mitigation Strategies

Adverse change mitigation strategies to prevent performance degradation or outright failure of the ACAPS need to be in place before operating in a highly semi-automated or fully automated mode. Upstream changes that can affect the performance of the ACAPS need to be identified as early as possible and communicated to the ACAPS maintainers.

Changes to the Decision Inputs

When the input data for decisions are changed, there are different approaches depending on the type of change. Common types of decision input changes are listed in this section, as well as recommended strategies for mitigating, as follows:

- **New values added to input**. An example of this would be the addition of new equipment IDs or a new organization. There is nothing in the historical data that matches the new data, and any attempt to use these data to automate should be considered suspect. In this case, the ACAPS should either automatically (through input anomaly detection) or manually (through rules) refrain from automating records using these values until enough training data have been generated and the models can be retrained.
- **Values changed in input**. An example of this would be a new equipment naming convention or an updating of organizational names. In this case, there are historical data that match these data, but without additional intervention the ACAPS will be unaware of this. These changes can be handled the same as new values added to

[35] https://en.wikipedia.org/wiki/Multi-armed_bandit.

input without issue, but the opportunity cost of missed automations may be large depending on the percentage of records with changed values. Another option would be to map the new values to old values during ingestion into the ACAPS and make decisions based on the old values. This would keep the ACAPS running near full efficiency at the cost of some changes on the input integration.

- **Input fields removed**. In some cases, certain input fields will cease to exist or be filled out. This can happen as a result of a change to the CAP source system or as a procedural change to no longer require certain information. In these cases, it is very important to retrain the ACAPS models on these historical data with the missing input fields removed. Not doing this and instead feeding empty values for new CRs can result in unexpected model performance and has a high chance of a negative impact to ACAPS accuracy.

Changes to the Rules/Processes for Making the Decision

In this case, the relationship between the input data and the decisions is being fundamentally changed. Examples of this could be a management decision to increase the evaluation level of effort for all significant injuries or a reclassification of which systems are safety-related. When this happens, it is usually the case that most of the historical data are still useful, but records representing the new rules need to be gathered or generated to train the ML models on. The following techniques can be leveraged in this instance:

- Increase manual sample rate and confidence thresholds for a short period to not automate as many records, and retrain the AI models with the new examples to learn from.
- Go back to historical CRs, and re-review some of them with the new rules; then, train the models to learn with these updated historical data.
- If the changes are to a subset of records that can be determined with a simple rule (for example, any CR with equipment from certain systems, CRs containing the word *injury*), update the CAP automation system to bypass automation for all records matching this criterion until enough training data are gathered.

Changes to the Decision Labels

If the decision labels are changed, the CAP automation models will no longer be predicting the correct labels. If the new labels have a one-to-one relationship with the old labels, you can use a process to translate the old predictions into the new labels. In most cases, however, there are major differences between the labels that do not follow such a simple mapping.

Strategies for the different types of mappings are listed here. If the new labels include a mixture of the following rules (for example, the change from the 2013 INPO performance objective and criteria codes to the 2019 WANO codes), multiple strategies can be combined:

- **One-to-one**. Recode historical data or translate the output of the model.
- **Multiple-to-one**. Multiple labels in the old system have been combined to one label. The strategy is the same as the one-to-one.
- **One-to-zero**. The label has been removed from the new system. For multilabel problems, simply remove the label. For multiclass problems, the historical data must either be removed or recoded to match one of the new labels as closely as possible.
- **One-to-multiple**. One label from the old system matches multiple new labels. This usually happens when more specificity is added in the labeling. For multilabel problems, the historical examples can be re-translated to add all the matching labels; then, new data will be gathered through manual sampling. If this is inappropriate or if the problem is multiclass, either some amount of the historical data must be recoded or new data must be collected with manual processing.
- **Zero-to-multiple**. A new label exists that did not exist at all in the old system. This is the most challenging case to handle. The strategies for the one-to-multiple situation are still applicable, but all records need to be evaluated for re-labeling or manual processing because there are no historical data to limit which records need the data.

Section 11: Long-Term Impacts to Other Plant Activities

Transitioning to a semi-automated or a fully automated CAP screening process has long-lasting impacts to other interfacing processes at an NPP. The transition to, and operation of, such a system must be managed carefully to ensure that impacts to other plant activities remain unchanged, benefit positively, or are otherwise weighed and managed against the positive outcome of automating CAP screening. Similarly, changes made to other process interfacing with the ACAPS must be managed so that ACAPS is not inadvertently negatively impacted. The following sections cover details of various plant process interfaces to an ACAPS.

Plant Process Interface

The CAP screening process at many NPPs can be visualized as a consumer-producer model in which certain business inputs are consumed and product is produced for downstream consumption for other business purposes. Figure 11-1 shows examples of business data producers that feed an ACAPS and business consumers that intake data from the ACAPS.

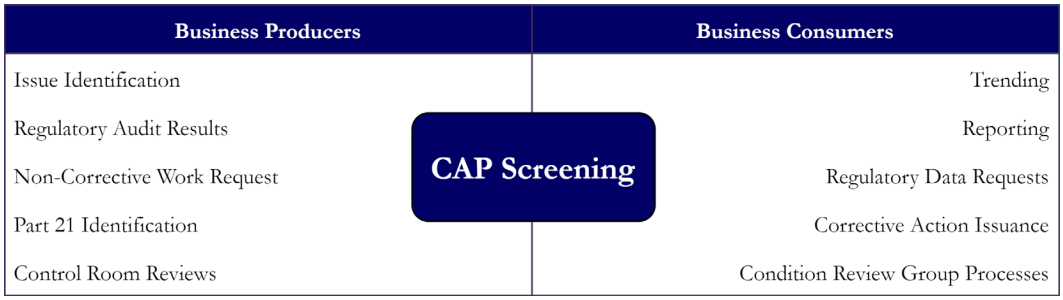


Figure 11-1
CAP screening inputs and outputs as a producer and consumer model

ACAPS implementers should be aware that these processes, both consuming and producing data, may not be immediately obvious. Changes made to a producer process that seem simple to a human can have adverse effects to an ACAPS. More apparent examples of these changes may be items such as an alteration of the software that employees use to fill out CRs, a change in the way regulatory audit results are documented, how NRC Part 21 Reports are handled, and potential changes in the way issues are routed to control rooms for review.

From the consumer's perspective, CAP automation is less impactful. Most systems will not be directly affected because first iterations of ACAPS will simply replace portions of the CAP screening data feeds in a like-for-like fashion. Downstream consumers that must intake data from the ACAPS process will be affected by the quality of the data produced by the ACAPS. In implementations without model confidence considerations, ACAPS will not be as accurate on subsets of the results as humans. Downstream consumers will need to account for this difference by adapting processes to account for the change in output. For example, an ACAPS producing corrective actions with an owner group may fail to differentiate between very closely related owners (for example, Mechanical Maintenance Team A versus Team B). In another example, a downstream CRG or management review committee may see more CRs with incorrect fields and may find that they need to manually send more CRs back for a second screening. Processes for handling these errors need to be developed and accounted for so that the rate of error and cost of error handling do not exceed the gains in efficiency ACAPS brings elsewhere.

More subtle changes can be induced by seemingly unrelated procedure changes, procedural process changes, and human resource or organizational system changes, among others.

Procedure Change Considerations and Guidance

After an ACAPS is in place, the process for changing procedures is no longer as straightforward as it used to be. When making changes to procedures that alter the data inputs, outputs, or decisions in the CAP process, it now becomes critical to be aware of downstream or upstream impacts to the ACAPS. Previously, without an ACAPS in place, procedure changes needed to be communicated only to screening personnel and they could adapt to the change relatively quickly. Under an ACAPS, changes to the expected process cannot be adapted as quickly and must be known further in advance. In addition, unknown changes made can cause an outright failure of the ACAPS or subtle performance degradations that affect the quality of output.

To mitigate the impact of procedure changes to CAP processes, personnel need to be aware that they are making a change that could affect the ACAPS. Awareness can be developed in the following ways:

- **Communication that an ACAPS is in use and the limitations of such a system**. Everyone interacting with the ACAPS—even CR initiators—should be aware of the existence of the system and its potential impact.
- **Use of impact forms for procedure change**. Impact forms should be generated by the procedure change initiator, and the ACAPS owners should be included in the impact communication. This ensures that performance impacts can be identified and mitigated.
- **ACAPS maintainers or business process owners should identify plant procedures or guidelines that govern upstream processes**. These procedures should be reviewed for steps or sections that govern, interact with, or could impact the ACAPS. A notation in the document should be made to indicate that changes made to that section could adversely impact the ACAPS. The process of doing this review and notation addition should become ingrained into the ACAPS change process so that, as elements of the ACAPS are enhanced or changed, new impacts or dependencies are documented.

Key Operating Experience

Procedural Impacts at Utility G.

During implementation of the ACAPS (and other automation projects) at Utility G, the potential impact of upstream procedural changes to newly automated systems was recognized and mitigated. The information technology (IT) department's change process was revised to add a step requiring the automation owner to review and revise plant procedures that may use or govern the automation. Notations were added in the plant procedure basis documentation so that an employee performing a procedure revision with no prior knowledge of the automation will always become aware of the impact to the automation system and issue a change impact form to the correct automation owner.

Data Changes Outside CAP

Changes to data outside the CAP process, but that are used by the ACAPS, can have an adverse effect on the ACAPS. There are many ways this can happen, including changes to plant equipment designations, naming conventions, and so on. Perhaps the most common example is changes to organizational structure.

Organizational changes—specifically changes in organizational unit identifying field (unit or department name, org unit id, and so on)—can adversely affect the performance of an ACAPS. For example, a simple change such as changing the org unit *Mechanical Maintenance B* to *Valve Maintenance One* could result in a shift in input data causing incorrect

predictions. This type of occurrence is especially challenging because it is pervasive, occurs somewhat frequently as organizations change, and, depending on the specific ML technique underpinning the ACAPS, may not be easily, quickly, or automatically relearned.

This type of change would more generally be described in ML practice as concept drift but in this case occurs instantaneously as the organizational update is made. Other examples of this type of change would include employee legal name changes, equipment noun name changes, or the change in a downstream work product type. Mitigation strategies for concept drift exist, and discussion can be found previously in this report. In general, an ACAPS operator should be aware that any sudden change in a reference to any piece of used information can cause adverse consequences to the ACAPS, and best effort should be made to become aware before such change is made live.

As covered, sudden changes made to plant references, taking again the case of an organizational unit rename, will cause varying effects through the ACAPS. The magnitude of this effect on the following specific components of the ACAPS will vary, depending on the specifics of the component implementation:

- **Data.** High impact; field names or fields themselves change or may cease to exist in the source system.
- **Data pipelines/extract transform load**. High impact; potentially a breaking change. ACAPS or intermediary systems that transform the source data into different forms are usually written to expect the existence of a field in a data set and may completely fail to operate when a field or fields no longer exist. Changes made to field values are lower impact because a well-designed pipeline or extraction code will be able to handle a missing or new field value within a certain field.
- **ML model**. Mild impact; performance degradations would be expected, the severity of which depends on the feature importance of the changed field or field value. Though ML models will expect to see the same fields, many/most different types of ML models will be able to handle missing, new, or altered field values without outright failure. Missing fields will have a very high likelihood of being caught in necessary data pipeline operations prior to the ML model step, and, therefore, the ML model step is less exposed to that type of change and the errors it may cause. Depending on the implementation architecture, the ML model can even self-recover from immediate concept drift. This would be the case in an online learning implementation.
- **ACAPS performance metrics**. Low impact; some metrics associated with missing fields may fail to update and appear frozen in time. Some metrics for new or changed fields may experience previously unseen volatility until a sufficient number of records with the new data have been processed and incorporated into the calculation.

Impact to Plant Metrics

NPPs adopting an ACAPS are making substantial changes to the way their CAP data are generated; one of the key considerations is the potential impact on downstream process metrics. Besides recognizing the impact on some of these process metrics, it is important to recognize what the metrics were originally intended to measure and assess whether the ACAPS is actually having a positive or negative impact on these items.

Although this is not an exhaustive list, the following are examples of metrics that could be affected by an ACAPS:

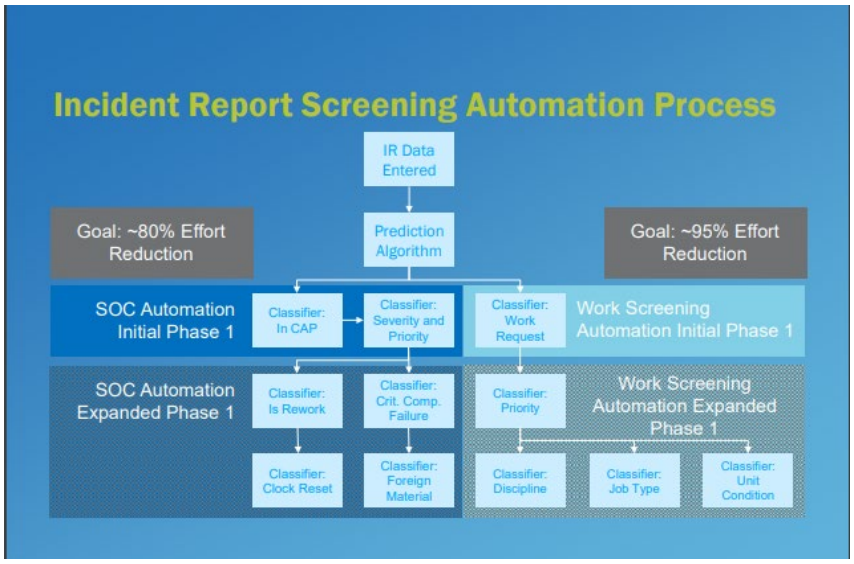
- **Screening committee throughput per person-hour**. After adoption of an ACAPS, it would be expected that screening throughput per person-hour may actually decrease. In this case, it is important to recognize that raw effectiveness has not actually been affected but rather that the easier CRs have been automated and the average time per CR screening will be higher on the more challenging CRs still being manually screened.
- **Work order cancellation rate**. One side effect of an ACAPS that generates work orders can be an increase in work order cancellation rate. ACAPSs generating potential corrective actions sometimes elect to over-generate potential actions and rely on CR assignees to remove the less valuable actions. This is done because it is often easier to remove an unnecessary record than create a new one from scratch. In cases in which the ACAPS functions this way, it is critical to communicate and ensure that stakeholders understand the impact to cancellation rates.

Section 12: State of the Industry Survey

The following sections represent survey data as collected from utilities known to have ACAPS in production, in development, or in conceptualization phases—and from participating national laboratories and other nuclear industry organizations. Survey data were collected through an e-mail questionnaire, if responses were returned, and otherwise through compilation of publicly presented material. All survey material is aggregated into Approach, Current Status, Improvement Opportunities, and Plans for the Future sections.

Utility A

Utility A has approached CAP automation with projects in two areas: automated CAP screening and automated MRFF. The stated goal of the automated CAP screening project is to achieve an 80% reduction in the effort required to screen CRs, executed in two phases. The first phase is a CAP classifier in which the severity and priority fields are determined and corrective actions automatically generated. The second phase expands the first phase into more discrete targets, where severity and priority are further classified into categories and generated corrective actions are assigned specific fields related to the corrective action (see Figure 12-1).



SOC = screening oversight committee

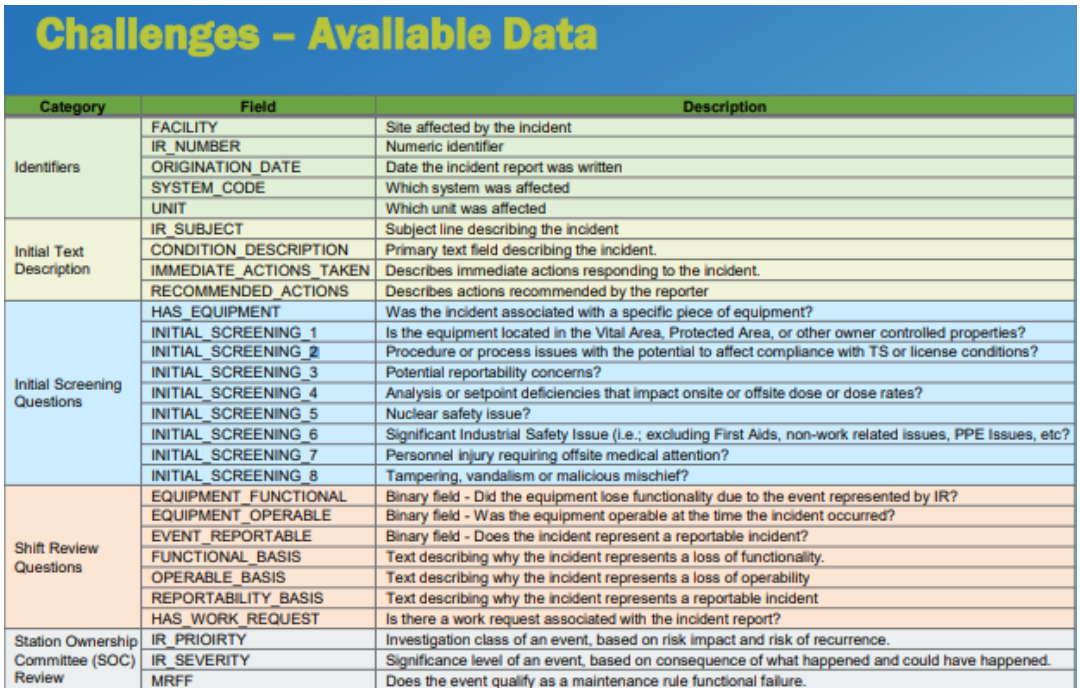
Figure 12-1
Utility A screening automation implementation flowchart[36]

Approach

As covered, the approach taken in the project was in phases, first completing more broad portions of the ML project portion and then later moving into fine-grained classification. From early in the project, Utility A held a focus around explainability of the models—that is, there should be some available reason as to why one output was selected. It also identified key areas of the automation process that are required to achieve the stated automation goals, which become the target variables for prediction by the ML models. The target variables are SCAQs, identification of critical component failures, nondiscretionary clock resets (human performance issues), and rework.

[36] NRC AI/ML Workshop, https:/www.nrc.gov/docs/ML2132/ML21326A192.pdf.

For the technical aspects of the work, Utility A's innovation group partnered with Vendor A and National Lab A for both automated screening and automated MRFF. The technical approach to each modeling problem was similar because the data sets and contextual information around each decision are similar. Utility A's historical data related to making these decisions displayed significant class imbalance. The CRs deemed significant—a Severity 1, 2, or 3 label—made up less than 1% of the approximate 410,000 CRs documented from 2017 to present. The training data have several fields, of which seven are free-form text. The remainder are categorical or numerical fields, typical of most NPPs. Atypical is the number of initial screening questions available in the initial data after the CR has been generated by the initiator (see Figure 12-2).



Challenges – Available Data

Category	Field	Description
Identifiers	FACILITY	Site affected by the incident
	IR_NUMBER	Numeric identifier
	ORIGINATION_DATE	Date the incident report was written
	SYSTEM_CODE	Which system was affected
	UNIT	Which unit was affected
Initial Text Description	IR_SUBJECT	Subject line describing the incident
	CONDITION_DESCRIPTION	Primary text field describing the incident.
	IMMEDIATE_ACTIONS_TAKEN	Describes immediate actions responding to the incident.
	RECOMMENDED_ACTIONS	Describes actions recommended by the reporter
Initial Screening Questions	HAS_EQUIPMENT	Was the incident associated with a specific piece of equipment?
	INITIAL_SCREENING_1	Is the equipment located in the Vital Area, Protected Area, or other owner controlled properties?
	INITIAL_SCREENING_2	Procedure or process issues with the potential to affect compliance with TS or license conditions?
	INITIAL_SCREENING_3	Potential reportability concerns?
	INITIAL_SCREENING_4	Analysis or setpoint deficiencies that impact onsite or offsite dose or dose rates?
	INITIAL_SCREENING_5	Nuclear safety issue?
	INITIAL_SCREENING_6	Significant Industrial Safety Issue (i.e.; excluding First Aids, non-work related issues, PPE Issues, etc?
	INITIAL_SCREENING_7	Personnel injury requiring offsite medical attention?
	INITIAL_SCREENING_8	Tampering, vandalism or malicious mischief?
Shift Review Questions	EQUIPMENT_FUNCTIONAL	Binary field - Did the equipment lose functionality due to the event represented by IR?
	EQUIPMENT_OPERABLE	Binary field - Was the equipment operable at the time the incident occurred?
	EVENT_REPORTABLE	Binary field - Does the incident represent a reportable incident?
	FUNCTIONAL_BASIS	Text describing why the incident represents a loss of functionality.
	OPERABLE_BASIS	Text describing why the incident represents a loss of operability
	REPORTABILITY_BASIS	Text describing why the incident represents a reportable incident
	HAS_WORK_REQUEST	Is there a work request associated with the incident report?
Station Ownership Committee (SOC) Review	IR_PRIOIRTY	Investigation class of an event, based on risk impact and risk of recurrence.
	IR_SEVERITY	Significance level of an event, based on consequence of what happened and could have happened.
	MRFF	Does the event qualify as a maintenance rule functional failure.

Figure 12-2
Utility A available fields in historical data[37]

For automated screening, the ML techniques involved a mix of Naive Bayes models and artificial neural networks. CAP data for four years from several different stations were used to train the models (roughly 600,000 records). The textual fields were first split into one, two, and trigrams processed by a Naive Bayes classifier for each target label and recombined with the categorical values, all of which was then input into a simple feed-forward neural network. The output of this network was used to determine the category of the final target label. Confusion matrices were constructed for each of the target labels to measure system success.

[37] NRC AI/ML Workshops, https://www.nrc.gov/docs/ML2127/ML21277A139.pdf.

Utility A has also partnered with National Lab A on automated trending. This material is covered in the National Lab A portion of this section.

Current Status

The MRFF automation portion of the project is in production and operational as of late 2019. Monitoring of the system is in place, and retraining of the system has been occurring to add new data and increase performance.

The CAP screening automation portion of the project is still being developed and will begin a pilot in the first quarter of 2022.

Improvement Opportunities

No improvement opportunities were highlighted although raw accuracy as a measure was called out to be inadequate as a metric when determining the success or failure of a CAP automation model.

Plans for Future

Utility A plans to continue the CAP automation pilot into 2022, expanding to between two and four different stations until an undefined level of enterprise confidence in the system is obtained. It plans to explore alternative uses for the project around NRC inspections and to eventually deploy open-source tools for broader industry use. In addition, Utility A is developing a NewCap 2.0 system, part of which is adding a user interface into its ACAPS.

The MRFF automation project will be expanded with a user interface/ results page.

Utility B

No response was received from Utility B, and no material was made publicly available.

Utility C

No response was received from Utility C, and no material was made publicly available.

Utility D

A nuclear plant at Utility D generates approximately 9000 CRs each year with a screening cost of $1.01 million per year, equating to roughly $112 per CR.

The plant screens CRs for the following fields:

- Priority (CAQ and SCAQ, or non-CAQ)
- Severity (risk significance)
- Owning department/resolving process

The screening process is performed by a CRG of five or six people four times per week with input from 20 to 30 external people. A management review team meets three times per week to review results issued by the CRG. The plant developed an ACAPS in mid-2020 to provide recommendations to the condition screening group.

Approach

The plant partnered with IBM to use its IBM Watson Cloud AI product to begin automating CAP screening at a cost of $250,000 initially and $125,000 per year thereafter, reducing the total cost to screen a CR to approximately $13.00. CR data are preprocessed before being processed by Watson to identify certain specific features in the data, such as limiting conditions of operation or operability concerns. Part of training on Watson involved indicating exactly which keywords or phrases in the CR description were associated with certain classification targets; 750 CRs were annotated with these keywords and phrases by the team over a two-month period in early 2020.

The system was initially trained on 750 historical CRs. Pilot accuracies were 82.8% for priority (CAQ/not CAQ), 75.9% for severity (multiclass field), and 62.1% for both fields. After additional training and feedback, accuracies were 96.9% for priority, 83.1% for severity, and 80.0% for both fields. The automation recommendations are added to the screening review report where they are reviewed during the routine review processes. Human added fields are also included, and Watson's applied fields are included for comparison.

Current Status

The plant's ACAPS is currently in production, providing recommendations directly to the CRG. Inception of the system occurred in November 2019, and development and refinement proceeded until August 2020. In August 2020, the plant's CRG began using outputs of the

system to grade by exception. The plant used the solution to eliminate CR pre-screening, CRG pre-screening, the CRG meeting, and post-screen processing tasks.

Improvement Opportunities

Improvements in accuracy rate for the Owners Group field were noted as lower than the rest and are currently being researched for improvement. Improvements can include alteration of business processes to better work within Watson's capabilities.

Plans for Future

The plant hopes to realize an 85% reduction in total cost of CAP screening processes. Trend coding of each CR is a current area of research because, when the CRG disbands, initial trend codes will need to be applied to the CRs.

Utility E

Utility E developed a pilot ACAPS in mid-2020. The pilot project expectation was partial automation of the CAP priority field. Utility E desired to build in-house expertise and knowledge and therefore decided not to use any vendor support in this effort. All development was performed by in-house staff. The project was funded as part of an innovation activity under operations and maintenance expenses with the objective of increasing screening efficiency.

Approach

Utility E established a small internal team of data scientists to develop a pilot CAP automation system in 2020. The team used techniques from traditional NLP for processing natural language from the CR into consumable features. These techniques were text cleaning and spelling correction, Levenshtien distance, and stemming of words. Naive Bayes, logistic regression, and random forests were used for classification modeling, and a rules-based model was applied on top of these results. The data available to train the model had high class imbalance.

Current Status

The pilot was completed in 2020, and results were mixed. Although some ground was made, the accuracy levels were not sufficient to automate. Test data accuracy was ~79%. The best model discovered was to apply an initial rules-based approach and then layer on an ML-based technique.

Improvement Opportunities

Improvement opportunities included exploring the trade-offs between over- and undersampling records to correct for class imbalance or using various synthetic data generation techniques to generate additional training data. Additional plans may include alterations of the business process to allow for more structured data. This would include efforts such as altering the CR input form to allow the user to select more discrete choices.

Plans for Future

No further plans were noted.

Utility F

No response was received from Utility F, and no material was made publicly available.

Utility G

Utility G established a dedicated data science team in 2017 and used internal resources and open source tools to develop its CAP automation system. Utility G began with an automated trend coding system, then CAP screening automation, and finally MRFF screening. Utility G processes about 20,000 CRs per year.

Approach

Automated Trend Coding

Utility G applies trend codes exclusively using INPO/WANO performance objective and criteria codes, with minor exceptions for specific human performance error codes and clock resets. The team's approach was to treat trend coding as a multiclass classification problem, using textual and categorical features related to the CR. For automated trend coding, textual features were processed using traditional NLP techniques, and words were one-hot encoded and combined with categorical features. This input was fed into a simple feed-forward neural network model, and the applicability of the performance objective and criteria codes was ranked by model probability. A multi-armed bandit approach was used to further choose the selection category for the applicable codes, depending on user preference. Several options were tested, and a Top-5 strategy was selected. In later iterations of this model, textual feature modeling was refined using large-scale word embeddings and more advanced sequence-based neural network models, and target selection was further modeled with approaches similar to sequence-to-sequence modeling to allow a variable number of targets to be selected.

Automated Condition Screening

Utility G's approach to automated screening was to individually model each of the required target fields needed to advance a CR through the human process. Utility G requires the significance, responsible group, and any corrective actions to be applied prior to leaving the CR screening step. The team paid additional attention to the accuracy of the Significance field because a fully automated false-negative result (an adverse condition being declared *not adverse*) would be extremely detrimental to the project.

The team modeled each of these fields independently and with separate models. Available historical CR data consisted of primarily textual data, with some categorical data describing equipment and personnel. The modeling work consisted of development of various NLP-focused neural network architectures implemented in Python using both the *Keras/Tensorflow* and *PyTorch* libraries and with gradient boosting methods as needed. For the severity level target, the team treated the target as a binary outcome. The historical data available were severely unbalanced with the majority class being the condition not adverse to

quality. For the responsible group target, the team treated it as a multiclass outcome and selected the highest probability result. Selection was made from approximately 325 different classes. For the corrective actions, the team tried two approaches. The first approach independently modeled each of the fields required on an individual corrective action independently. The first set of fields among all independent models was selected as a corrective action, the second set as the second, and so on. The second approach considered corrective actions as a chain of conditionally dependent events and modeled as such.

Each of these models was prepared behind a serving API, and a common API was used to wrap the models into a common product. The common API was served in a highly available *Kubernetes* cluster, and integration hooks were inserted into the station's main database so that predictions could be made on demand when a new CR was generated.

MRFF

At Utility G, the Maintenance Rule program is implemented by strategic engineers who perform a review of every CR with an equipment failure to detect potential MRFFs. This review was performed through an existing web application that presented daily CRs organized by plant system. Utility G determined the most appropriate initial solution to be a recommendation system that presented the model results on the existing MRFF screening web application.

The MRFF determination process at Utility G had three potential outcomes: MRFF Yes, MRFF Indeterminate, and MRFF No. In the case of MRFF No, an additional subset of potential justification categories was provided. Based on this understanding of the MRFF process, a binary classifier was developed to predict either MRFF Indeterminate or MRFF No.

The binary classifier model was trained on CR text and categorical features from both the CR and plant equipment associated with the CR. One of the early challenges with this system was understanding what historical data were available to plant personnel at the time the MRFF decision was being made because the human-based MRFF determination timeline is not consistent within the data set. In the case of this system, particular care was required in the historical data gathering stage so that the appropriate data were truncated prior to ingestion into a model training loop.

This model was developed as a neural network with the *PyTorch* library. The interface with the model was delivered with two web-based APIs: one to look up all required CR inference data given a CR number and one to perform the model inference given the required CR data. This system allowed front-end consumers to query the model using CR details.

Full automation of a subset of CRs for the MRFF determination step would require an additional prediction by the model. As previously indicated, MRFF No conditions at Utility G require a level of additional justification. This justification most often comes in the form of text-based categorical justifications such as Minor Deficiencies, Related to Administrative Controls, and Other. Therefore, to automate the MRFF No case, the model must also predict the categorical MRFF No label that includes the appropriate justification. At this step in the model iteration, the decision needed to be made to change the model from a binary classification model to a multiclass classification model or a more layered approach in which the initial binary classification still occurs with a multiclass classification inference in the case of MRFF No. In this application, it was decided to make the justification a secondary prediction because the accuracy required of the justification was not as high as the required accuracy for the MRFF No versus MRFF Indeterminate prediction.

Current Status

In the case of automated trend coding, the application of codes has been 100% automated since mid-2019.

In the case of automated condition screening, the system was run from fall 2019 to summer 2021 in a recommendation mode, after which the first of the full automation functions was switched on with conservative guidelines. About 20% of the total CR population is being fully automated while maintaining 100% accuracy on the severity level field. CAP program owners report that they are satisfied with the results and desire to allow a higher percentage to be fully automated.

In the case of MRFF, overall performance of this initial model would allow approximately 40% total automation of MRFF determinations through the binary classification system with virtually no expectation of an error. The output of this model was displayed as an informational label on the existing MRFF review web application to engineers to allow a review and feedback period before moving into a full automation state. MRFF review automation has been in place since late 2020.

Improvement Opportunities

Utility G desires to improve visibility and configurability of the CAP screening automation system and develop updates to AI models to leverage recent external innovations.

Plans for Future

Utility G plans to change the thresholds to increase the proportion of CRs being processed by the automated CAP screening system.

Utility H

Utility H is creating a cohesive CAP product that it believes will transform plant operations. The primary goals are to more accurately classify CAP issues with analytics, more quickly resolve corrective actions, and improve plant performance through created insights. It plans on integrating an ACAPS with various user inputs such as mobile apps and field devices and providing results back to plant staff through tools such as CAP Intelligence Advisor, which will provide a comprehensive view into the CAP data.

Approach

Utility H is currently developing its ACAPS in an agile fashion and plans to deliver features and functionality as soon as reasonably possible. Utility H has a combination of a small team of in-house developers supplemented by a larger team of outsourced developers. Utility H has been collaborating with National Lab A for analytics support.

Current Status

Utility H is currently in the planning phases of an ACAPS that will create recommendations for required CR screening fields and assign required corrective actions. Utility H plans to have a minimum viable product for an automated CAP developed and delivered in 2022.

Improvement Opportunities

No improvement opportunities have been identified at this time.

Plans for Future

The plans for the future of automated screening at Utility H include tight integration with the remainder of its CAP technology products. Upstream and downstream integrations include products such as CAP Intelligence Advisor, which is a custom interface to update and review CR information and linked actions, a notification engine for CR information changes, advanced CAP search capabilities, and reporting and trend analytics. Utility H expects a full minimum viable product of the CAP suite of products by the end of Q1 2022, beginning phase two of the project at the same time.

National Lab A

National Lab A is part of the U.S. Department of Energy's (DOE's) complex of national laboratories. The laboratory performs work in each of the strategic goal areas of DOE: energy, national security, science, and environment. National Lab A is one of the nation's laboratories for nuclear energy research and development.

Approach

National Lab A conducted research with CAP data from the U.S. nuclear power industry to determine the scalability, transferability, and deployability of AI models for use in CAP automation efforts. National Lab A used a combination of unsupervised and supervised ML methods coupled with NLP techniques to assign keywords/topics to CRs and enable automated trending for identification of topics. In addition, it integrated a live data link from a utility to pilot a data portal to facilitate data sharing between organizations interested in CAP data.

As part of the ML model development, National Lab A has developed a tool called *Machine Intelligence for Review and Analysis of Condition Logs and Entries* (MIRACLE). This tool performs document topic extraction through latent Dirichlet allocation (LDA), which is a statistical method to discover a fixed number of previously unknown links between sets of independent documents. The output of this tool is a common set of topics that appear to be common across a large number of common nuclear data (see Figure 12-3).

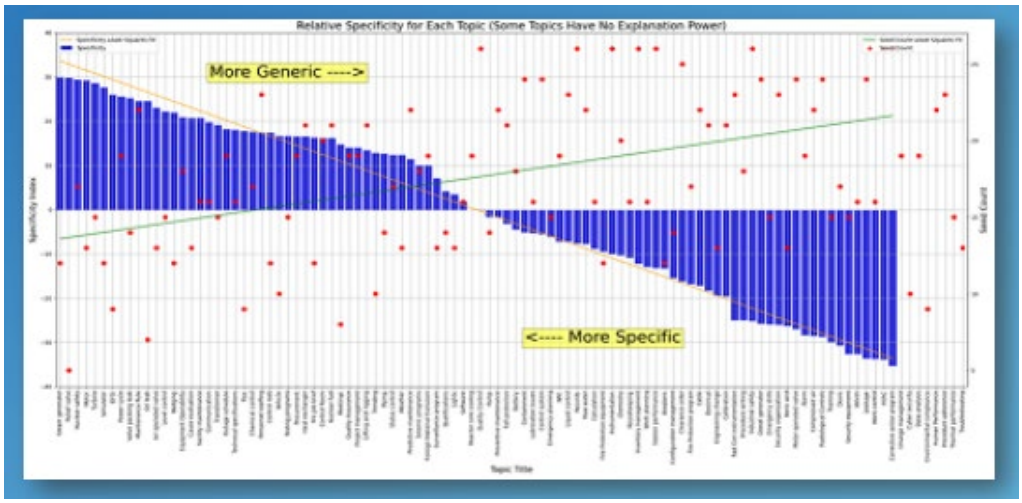


Figure 12-3
Example LDA topics as determined by the MIRACLE program[38]

[38] NRC AI/ML Workshops, https://www.nrc.gov/docs/ML2127/ML21277A139.pdf.

Current Status

The initial pilot is complete, and initial results indicate that the AI models developed can indeed be scaled across multiple sites and achieve better performance. The developed models can successfully identify themes and topics for CAP data.

Improvement Opportunities

Model transferability approaches were researched, including an easy-to-deploy approach improving on single utility models.

Plans for Future

National Lab A plans to conduct additional requirements analysis and customize the generated trends to meet the assessment and inspection requirements as well as automated data-driven decision making.

Section 13:Commonalities in Success and Failure

Across the various operating experience gathered from utilities that have implemented CAP automation projects, there are some commonalities among successes and failures. Although correlation does not necessarily mean causation, it is worthwhile to highlight these commonalities.

Successes

High Level of Engagement with CAP Stakeholders

Utilities are spending between six months and two years proving that systems will perform as expected to build organizational confidence and increase adoption.

Use an Agile Approach

An agile approach is an incremental, iterative approach to piloting the project in the organization. A fast iterative approach allows rapid cycles of project work and stakeholder feedback.

Modeling Work Starts Broadly and Becomes More Specific

Project modeling work begins by predicting very broad targets and becomes more specific as the project progresses. This approach allows external resources to become acquainted with the tasks at hand.

Use of Advanced Modeling Techniques

Leverage more advanced modeling techniques and tools to drive increased accuracies, specific to the data being processed, including specific NLP techniques and ensemble modeling.

Use of Varied Sources of Data

Use as much varied data as possible because even nonutility-specific data appears to improve model performance.

Use of Model Probability Values and Set Thresholds

Bucketing prediction results into different probability/confidence buckets allows different actions to be performed by the ACAPS depending on model confidence. ACAPS projects can proceed without perfect modeling accuracy if model confidence levels are accounted for.

Failures

Although there were no specific commonalities across all failures, there were notable aspects that had mixed results.

Use of Only an In-House Team

CAP automation AI system development solely with an internal team has led to mixed results.

Reliance on Rule-Based Systems

Use of rule-based subsystems within the ACAPS.

Ignoring Unbalanced Data

Not developing specific techniques for coping with highly unbalanced data sets will lead to failure. The CR severity fields being modeled are in the range of 98% negative class to 2% positive class, depending on the NPP.

Use of Incorrect Success Metrics

Using the wrong metric to measure modeling accuracy. Seemingly sufficient raw accuracy numbers will mask other issues and lead to poor project outcomes.

Not Reviewing Carefully for Information Forward-Leak

Forward-leak of information in a training data set that is highly indicative of a target outcome will boost success metrics of an ACAPS in development and cause failure when run in a deployed system.

Missing

The following components of a project as covered in this report appeared to be missing from existing implementations and/or are not on project plans.

ACAPS Monitoring

Monitoring of the ACAPS is missing from all implementations. No utilities are monitoring the ACAPS as a core piece of enterprise software.

Specific Quality Control Methods

These are not apparent in any of the implementations. In most projects, quality control and review of predicted outcomes are reliant on human-in-the-loop systems in which almost all of the automated outputs of the ACAPS are being reviewed by human experts as a step in the existing CAP screening process.

Tooling

There appear to be few consistent themes for tooling across the surveyed NPPs. Although some utilities focused on open source tooling such as Python and R, others have leveraged vendor software and methodologies with success. There appears to be almost no commonality among deployment technologies across utilities.

Section 14: Recommendations and Advice

Based on the industry survey and anecdotal experience, the following recommendations are provided for utilities interested in adopting an ACAPS. Following these recommendations will guide an NPP toward a successful CAP automation implementation.

Recommendations

1. Use an incremental adoption framework approach to roll out a system automation, allowing the organization to build confidence in and adapt to business changes that are required for a successful CAP automation project.
2. Automate different components of a CAP in the following order. This order minimizes the operational and regulatory risk associated with an automation failure and generally prioritizes lower complexity, higher value use cases first:
   - Trend code automation
   - MRFF review automation
   - Reportability review automation
   - CAP screening automation
     - Safety significance/CAQ
     - Level of effort
     - Responsible group
     - Corrective actions/work item generation
3. Build strong cross-functional buy-in from IT, CAP management, the data science practice, and upper management. This is generally good advice for any project, but this is especially important for adoption of an ACAPS. The combination of business operational impacts, IT systems integration, automation complexity, regulatory risk, and efficiency opportunities is unparalleled outside major IT system replacement projects.
4. Consider partnering with experienced contractors or a vendor for development of an ACAPS. Although some solely internal teams have had success, ACAPS is more complicated than most data science and IT projects, and inexperienced teams will struggle to produce the

required automation accuracy and develop a robust enough system. In addition, the cost of developing the entire system from scratch will take a large amount of the potential return on investment.

5. Leverage modern ML and AI techniques. A key component of successful CAP automation is achieving a high level of accuracy. The type of heavily textual data prevalent in CAP benefits immensely from newer ML algorithms and techniques, disproportionately so when compared to other common data science use cases in the utility industry.
6. Ensure that the ACAPS has sufficient auditing and monitoring capabilities. An ACAPS without these capabilities has a high risk of failure, either from overall degrading performance or the inability to analyze a high-profile automation error. Appropriate auditing and monitoring capabilities are critical for withstanding regulatory scrutiny and internal stakeholder concerns as well as ensuring overall system performance.

Common Mistakes

CAP automation is complicated, and many mistakes and pitfalls need to be avoided during the development, implementation, and maintenance of an ACAPS. This section describes common mistakes that are likely to occur during ACAPS adoption. Many of these mistakes are significant, and much care should be taken to be aware of and mitigate their potential impact.

Using the Wrong Metric for Accuracy

Perhaps the most common mistake in ACAPS development is using the wrong metrics for measuring the accuracy of the ML models. There have been several occasions in the industry in which much interest was attributed to a pilot ACAPS, only to discover that a 90% accuracy number being discussed was basically meaningless for a data set that is a specific outcome 88% of the time.

Not Comparing to Human-Level Performance

When evaluating the effectiveness of an ACAPS, it is critical to compare to human-level performance numbers instead of evaluating the system performance in a vacuum. It is common to meet many stakeholders who are unwilling to adopt an ACAPS if there are ever any errors in the system's output. It is critical in these cases to reference hard statistics about human-level performance and accuracy obtained through audit or oversight processes. Often, the error rates in the manual process are higher than people like to think.

Not Using ML Techniques for Text

Training ML models on text data is generally more complicated and challenging than on nontext data, but it is essential for an ACAPS. The vast majority of useful information in CAP data are inside the text and ignoring or underleveraging those data will result in significant decreases in accuracy levels. Even worse than this would be electing not to use ML at all and attempting to make a rules-based automation system work.

Including Automated Records in Training Data

Including automated records in training data can have disastrous results. Preventing the use of previously automated records can be difficult to enforce, especially if the ground truth data sources used to train the model also house production data. However, if training data are acquired through another transformation layer, such as a data warehouse or feature store, enforcement could be carried out in source to target data extraction or transformations. In addition, enforcement in source code could be used. Encapsulate data extraction logic so that flags and exception records are respected by default. Most importantly, strong documentation of the exception field or flag and communication to new developers not to use flagged records are needed. In a procured system, one should validate that automated records are excluded from training by means of system design. In addition, when automated records are correctly excluded, it is important to ensure that the remaining manual sample records are weighted correctly for training.

Neglecting to Monitor or Evaluate ACAPS Performance

Even after the CAP process is automated, it is important to systematically measure and monitor ACAPS accuracy. Unfortunately, there are many ways ACAPS performance can degrade over time, and without monitoring it is challenging or impossible to catch degradation and mitigate it before more serious impacts to the CAP process are realized.

Section 15:Conclusions

Overall Opinion of Feasibility Given the Current Technology

CAP automation provides an excellent opportunity for NPPs to reduce administrative overhead and improve the consistency and efficiency of their CAP processes. However, this opportunity is not a simple endeavor and comes with some risks. NPPs will first have to tackle challenging AI problems to automate key decisions in their CAP. After solving these challenges, NPPs will then need to work through software integration, monitoring, auditing, ACAPS maintenance, and other change management and regulatory challenges. NPPs interested in taking advantage of this opportunity will need to have strong internal teams with data science and software engineering experience or will need to partner with experienced vendors offering services and software for CAP automation.

Overall, CAP automation is feasible with current technology and techniques and within the current regulatory and operating environment for domestic NPPs. Within the United States, several utilities have implemented ACAPS with varying degrees of scope, automation, and savings—and the consensus appears to be that CAP automation remains a value-added initiative.

Approximate Cumulative Cost of Implementation

Components of Cost

Based on research, the operating experience gathered for this report, and anecdotal experience, the estimated costs of developing an ACAPS fall into the ranges shown in Table 15-1.

Table 15-1
Components of cost

Component	Initial Cost *Low/Med/High*	Annual Maintenance *Low–High*
Project initiation and requirements gathering (per solution)	$1k/$10k/$50k	N/A
AI model training and testing (per model)	$10k/$35k/$125k	5–10%
ACAPS component development and testing (per solution)	$3k/$10k/$50k	10–25%
Integration development and testing (per solution)	$5k/$40k/$120k	10–25%
Deployment and monitoring infrastructure (per NPP)	$0/$9k/$25k	10–90%[39]
Implementation and change management (per solution)	$15k/$26k/$52k	10–25%

Maintenance Rule Functional Failures

Based on operating experience, an example implementation of this automation assumes a new project, with two AI models of medium to high complexity, and an integration into a web view (see Table 15-2).

Table 15-2
MRFF

Component	Estimated Cost
Project initiation and requirements gathering	$10k
AI model training and testing	$45k
ACAPS component development and testing	$5k
Integration development and testing	$15k
Deployment and monitoring infrastructure (amortized)	$3k
Implementation and change management	$15k
Initial total cost	**$98k**
Annual maintenance	**$12k**

[39] Many deployment and monitoring software packages are licensed annually.

Reportability Review

Based on operating experience, an example implementation of this automation assumes a new project, with two AI models of medium to high complexity, and an integration into a web view (see Table 15-3).

Table 15-3
Reportability review

Component	Estimated Cost
Project initiation and requirements gathering	$10k
AI model training and testing	$45k
ACAPS component development and testing	$5k
Integration development and testing	$15k
Deployment and monitoring infrastructure (amortized)	$3k
Implementation and change management	$15k
Initial total cost	**$98k**
Annual maintenance	**$12k**

Trend Coding

Based on operating experience, an example implementation of this automation assumes a new project, with one medium- to high-complexity AI model, and integration into a source and web view (see Table 15-4).

Table 15-4
Trend coding

Component	Estimated Cost
Project initiation and requirements gathering	$5k
AI model training and testing	$60k
ACAPS component development and testing	$10k
Integration development and testing	$15k
Deployment and monitoring infrastructure (amortized)	$3k
Implementation and change management	$25k
Initial total cost	**$118k**
Annual maintenance	**$16k**

Automated CAP Screening

Based on operating experience, an example implementation of this automation assumes a new high-complexity project, with four models of medium complexity and one model of high complexity, with integrations into a source EAM system and a web view or reporting system (see Table 15-5).

Table 15-5
Automated CAP screening

Component	Estimated Cost
Project initiation and requirements gathering	$50k
AI model training and testing	$265k
ACAPS component development and testing	$50k
Integration development and testing	$100k
Deployment and monitoring infrastructure (amortized)	$5k
Implementation and change management	$50k
Initial total cost	**$520k**
Annual maintenance	**$50k**

Estimated Savings Calculations

The estimated savings are highly site/utility dependent; it is recommended that member sites/utilities use the EPRI Business Case Analysis Model (3002019454) to determine estimated cost savings.

About EPRI

Founded in 1972, EPRI is the world's preeminent independent, non-profit energy research and development organization, with offices around the world. EPRI's trusted experts collaborate with more than 450 companies in 45 countries, driving innovation to ensure the public has clean, safe, reliable, affordable, and equitable access to electricity across the globe. Together, we are shaping the future of energy.

Program:

AI.EPRI

3002023821

EPRI

3420 Hillview Avenue, Palo Alto, California 94304-1338 • PO Box 10412, Palo Alto, California 94303-0813 USA

800.313.3774 • 650.855.2121 • askepri@epri.com • www.epri.com