INL/RPT-23-74439

Light Water Reactor Sustainability Program

Tools and Methods for Optimization of Nuclear Plant Outages

September 2023

U.S. Department of Energy

Office of Nuclear Energy

INL/RPT-23-74439

Light Water Reactor Sustainability Program

Tools and Methods for Optimization of Nuclear Plant Outages

Diego Mandelli, Ahmad Al Rashdan, Shawn St. Germain
Svetlana Lawrence, Norman Mapes, Congjian Wang, Joshua Cogliati

September 2023

Idaho National Laboratory
Idaho Falls, Idaho 83415

lwrs.inl.gov

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

Page intentionally left blank

ABSTRACT

Refueling outages are one of the most challenging phases in a nuclear power plant (NPP) operating cycle. Refueling outages are extremely costly for an NPP due to the large amount of required resources and because of lost revenue due to plant being off the grid. Outage durations have steadily decreased across the industry over that last few decades primarily due to improved planning and coordination, but there are still many plants that struggle to meet the performance metrics accomplished by other utilities. Schedule resilience is one of the issues. NPP outages require scheduling thousands of activities within 30 days on average. Outage staff begin working on the schedule more than a year ahead of time and make every effort to build a robust schedule. Despite detailed planning, once the outage starts, numerous emergent issues typically appear along with schedule delays requiring continuous replanning and adjustment. When schedule disruption occurs during an outage, plant staff make urgent efforts to recover but are often not able to maintain the planned outage duration. These outage delays can cost a utility several million dollars per day. Tools that could help outage schedulers create a more resilient schedule and allow them to optimally reschedule emergent work could significantly reduce outage delays. One key aspect of creating a resilient schedule is to have accurate estimates for activity duration. Another important outage scheduling capability is the ability to schedule emergent work with minimal disruption. The Optimization of Outage Activities project under the Risk Informed Systems Analysis Pathway (RISA) sponsored by Department of Energy (DOE) Light Water Reactor Sustainability (LWRS) Program focuses on developing tools and methods to support NPPs with outage schedule optimization. The goal of the outage optimization is the completion of all planned and emergent outage activities as fast as possible while maintaining the highest level of safety. This report describes the initial development of tools to support outage management that leverage computational and machine learning methods developed within other RISA and LWRS projects.

CONTENTS

FIGURES

TABLES

Page intentionally left blank

ACRONYMS

CP	critical path
CPM	critical path method
DEP	dependency
DOE	Department of Energy
EFT	earliest finish time
EST	earliest start time
INL	Idaho National Laboratory
LCO	limiting conditions for operation
LERs	Licensee Event Reports
LFT	latest finish time
LST	latest start time
LWR	light water reactor
LWRS	Light Water Reactor Sustainability
NLP	natural language processing
NPP	nuclear power plant
POS	part of speech
PVNGS	Palo Verde Nuclear Generating Station
R&D	research and development
RISA	Risk Informed System Analysis
TF	total float
U.S.	United States

Page intentionally left blank

Tools and Methods for Optimization of Nuclear Plant Outages

1. INTRODUCTION

Nuclear power plant (NPP) refueling outages represent one of the more challenging aspects of managing a light water reactor (LWR) facility. Refueling outages typically require completing over 10,000 scheduled activities representing several thousand work orders in around 30 days. Most utilities use numerous contract workers to support outage activities, adding to the complexity and cost. Since the reactor is taken offline to perform this work, the utility is not generating revenue during the outage. The cost of lost revenue can be up to $2 million per day. To complete outages as efficiently as possible, planning begins more than a year before the start of the outage. Significant effort is placed on creating a schedule that minimizes outage duration with all the required work to be completed. In addition to the known work being scheduled for the outage, there may be several hundred emergent activities added to the schedule during the outage. This added work is typically referred to as scope growth. Most utilities have a formal process to limit scope growth to only those activities that are absolutely necessary, but there is always some work that must be added. This added scope, combined with activities that have slipped from the original schedule requires daily schedule realignment. While the outage managers have nearly a year to create the initial schedule, they may only have a few hours to reschedule each day. This difficulty in rescheduling is one of the reasons outage durations almost always exceed the original plan. When a refueling outage has a significant delay, the cost is substantially increased due to the necessity for the contract workers to stay on-site longer. Outage contract workers often move from site to site during outage season (spring and fall) supporting multiple utilities and the extension of their contracted work is typically difficult and expensive.

1.1 Background on Outage Scheduling

Plant outage scheduling is a very complex endeavor which requires a large amount of data and complex tools. The required data comprise the following:

- Data describing activities to be performed, including the following information:
  - Each activity's duration
  - Required resources for each activity
  - Dependencies between activities (i.e., the set of activities that needs to be completed prior to starting the activity under consideration)
    - Note that dependencies may be very heterogenous (e.g., system logic driven, tech-specs driven, plant risk[1] driven, or based on plant conditions)
- Data describing available resources for each day of the outage.

Provided the data listed above, schedule optimization tools developed in this project are designed to lay out the outage plan daily which includes:

- The subset of activities to be completed each day
- Activity time window
- Plant personnel are required to perform each activity.

[1] Typically, plant risk considerations are modeled through the plant probabilistic risk assessment (PRA) model.

Scheduling optimization tools develop an outage plan so that overall outage duration is minimized, with all activities completed. This minimization process carefully balances (throughout the optimization algorithm): activity duration, activity dependencies, and available resources.

An example of a plant outage activity scheduling plan is shown in Figure 1. Each node of the graph is a single activity and the edges between nodes represent dependencies among activities. The example schedule plan consists of eight activities (labeled using characters ranging from A to H) and the shown graph structure indicates that *A* corresponds to the initial activity while *E* corresponds to the final one. Activities *F*, *B*, and *H* can start after activity *A* is completed. The entire process is finished once activity *E* is completed.

The predicted completion time of an outage activity is of a particular interest. The predicted completion time is calculated by first determining the critical path (CP) which is the longest path (temporally speaking) from the initial to the final activity. From Figure 1, it is possible to identify four possible paths from *A* to *E*:

- Path 1: A-F-G-E (path duration = 50)
- Path 2: A-B-C-G-E (path duration = 60)
- Path 3: A-B-C-D-E (path duration = 65)
- Path 4: A-H-E (path duration = 45).

Out of these four paths, the third one (characterized by the highest duration) is the CP; in other words, activities A, B, C, D, and E comprise the CP. This implies that if the actual completion time of these activities increases, then the overall outage schedule will increase as a result. On the other hand, if the actual completion time of the activities that are not part of the CP increases, the overall outage duration may not increase. Thus, from an outage management perspective, the activities on the CP are typically closely monitored for potential delays and plant resources are reallocated if unexpected events occur.

Other elements of interest, which are calculated after the optimization process, are how early/late an activity can start/finish. The values are often indicated as earliest start time (EST), latest start time (LST), earliest finish time (EFT), and latest finish time (LFT). Last, for activities that are not part of the CP, the parameter *total float* (TF) is calculated. This parameter indicates how much an activity can be extended before it becomes a part of the CP (see Figure 2). Similarly, for the activities that are part of CP, the parameter *drag* is calculated. This parameter indicates how much such activity can be reduced before it gets moved out of the CP (see Figure 2). In summary, all activities indicated in the schedule shown in Figure 1 are described using the convention shown in Figure 3.

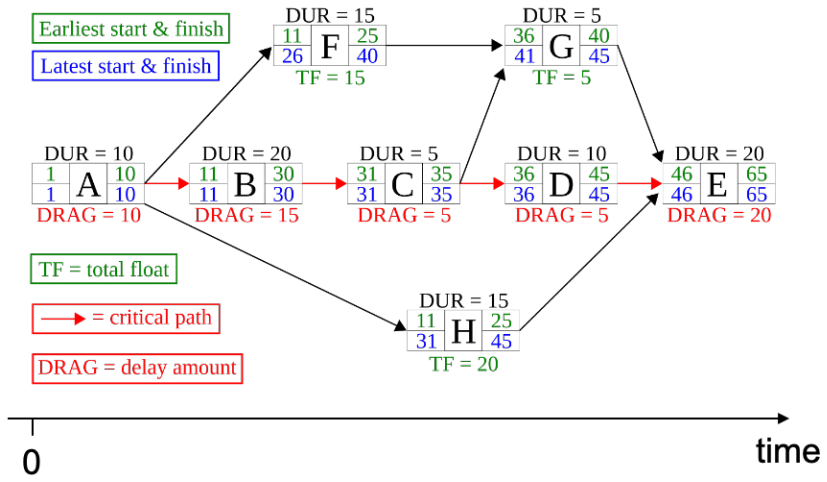

Figure 1. Example of plant outage scheduling plan(source:https://en.wikipedia.org/wiki/Critical_path_method).

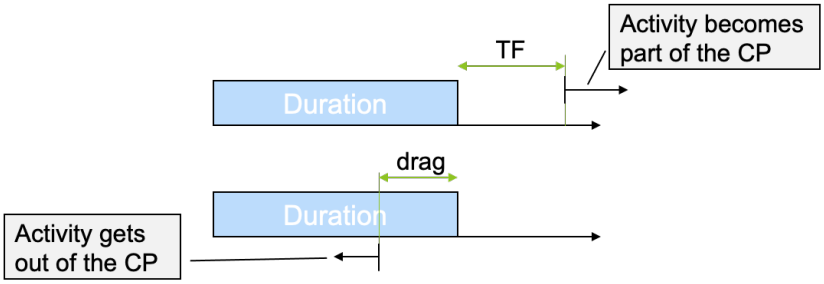

Figure 2. Graphical representation of TF and drag associated with an activity.

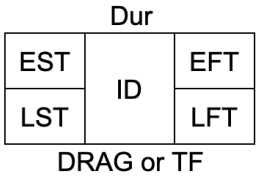

Figure 3. Summary of the defined and quantified parameters associated with an activity.

The process of outage scheduling planning and optimization described above, also known as the critical path method (CPM), is fairly mature and it has been widely used in the nuclear industry. However, a few criticalities can emerge when this approach is applied in a real context:

- The duration of activity is typically considered a point value while in reality, the actual duration is a variable based on past operational experience. The uncertainty in activity duration can be expressed in various ways (e.g., by providing):
  - The average duration value accompanied by a measure of its variance
  - The range of possible duration values is bounded by the observed minimum and maximum duration.
- There are multiple sources of duration uncertainties: number of people and skills of the assigned crew, operational conditions (e.g., weather), time of the day when activity is performed.

- The duration of an activity might be affected by the emergence of an event (that can be stochastic in nature) which needs to be addressed prior to the completion of such activity. This can have an impact on the actual activity completion time.
- New activities can materialize once the outage has started. The emergent activities must be incorporated into the schedule, including dependencies with other activities.

1.2 Objectives

The objective of this research project is to develop methods and tools to support plant staff in creating an outage schedule that has a high probability of completion in the desired timeframe. The research team interviewed outage managers and staff from three utilities. Each interviewee described similar problems experienced at their plant and all thought that better tools for evaluating outage schedules are needed. To improve on the current CP methodology used by most utilities as described in the previous section, we are investigating a method to calculate schedule resilience by creating a model of the schedule.

A resilient schedule is one that has been analyzed and adapted to account for the duration uncertainty capable of reorganizing activities to better absorb duration variability. A resilient schedule also presents margins for highly uncertain, non-CP activities. Last, a resilient schedule should have the capability to absorb the expected amount of scope growth without a significant disruption to the planned duration.

The tools for outage scheduling could be used in two main phases of the outage cycle. In the outage planning phase, the schedule optimization tool would help staff identify activities with a planned duration significantly different from historically witnessed durations. It would also point out the non-CP strings of activities that have high uncertainties in their durations and therefore have a high probability of becoming the CP. Identifying these high-risk strings of activities and recommending alternative schedule options to reduce the overall schedule completion uncertainty will help improve the overall schedule resilience. If some highly uncertain strings of activities must remain in the schedule, the tool would highlight them so that outage staff can maintain proper focus and oversight of those specific tasks to improve the chances of on-time completion. During the outage execution phase, the proposed tool would make recommendations on the best placement of emergent work activities on the schedule to maintain the highest level of resilience and to minimize the chances and magnitude of outage extensions.

In order to effectively model schedule resilience, we need activity duration uncertainty information along with the usual planned activity duration that is currently used in the CP methodology. In this case, each activity is assigned a duration distribution rather than a simple duration estimate. Various machine learning and artificial intelligence (ML/AI) methods will be investigated to automatically assign activity duration distributions based on the analysis of historical outage performance data. In cases where data are not sufficient to assign a duration distribution, a schema will be developed of a standard distribution to the assigned duration based on generic average completion time distributions for common types of work activities such as valve refurbishment, erecting scaffolding, circuit breaker refurbishment, etc.

While the concept of schedule resilience is understandable, specific metrics will need to be developed based on the schedule modeling to allow for automated optimization and recommendations. These metrics for schedule resilience will assist the outage schedulers in visualizing potential issues with the planned schedule and provide information useful in evaluating alternative scheduling options.

2. ADVANCED OUTAGE SCHEDULE MODELING

As indicated in Section 1.1, current outage schedule optimization methods rely on an activity duration expressed as a point value. However, the duration of an activity can be affected by many factors that can be either internal (e.g., number of personnel performing the task, plant crew workload) or external (e.g., discovery of a component failure during the inspection). Representing the effect of these factors on the activity duration via a single point value might have a major negative impact on schedule management during the outage.

The ability of a planned outage schedule to withstand a delay in the completion time of an activity (indicated with the term *robustness*) or to be able to counteract a change in an activity completion time (indicated with the term *resilience*) significantly increases the probability of the outage completion as predicted during the planning phase. From a plant operational standpoint, this also has an economic impact since, on average, each day of lost production (e.g., caused by outage delay) costs up to $2M in terms of revenue. While the estimation of activity duration values from past outage data is presented in Section 3, this section focuses on how uncertainties associated with activity completion time can be used to measure robustness and resilience of an outage schedule.

As a starting point, we need the ability to: (1) propagate activity duration uncertainties through an outage schedule (here indicated as *CP uncertainty*), and (2) evaluate how uncertainties associated with an activity duration affect CP completion time, activity drag, and activity TF. The propagation of activity duration uncertainties can be performed by assigning a probability distribution function (pdf) to an activity duration instead of reliance on a point value. This uncertainty quantification can then be easily performed through a classical Monte-Carlo sampling method. Note that when activity duration uncertainties are propagated through an outage schedule, the actual structure of the CP might change depending on the chosen activity duration values. In other words, depending on the sampled activity duration values, the sequence of activities that are part of the CP can differ. As an example using the simple outage schedule shown in Figure 1, if the duration of activity G becomes 15 (instead of 5), then the CP becomes A-B-C-G-E with a CP duration set to 75 (instead of a CP equal to A-B-C-D-E with a CP duration set to 65). Hence, when propagating activity duration uncertainties through an outage schedule it is relevant to track the set of possible CPs, their likelihood of occurrence, and their duration uncertainty.

The estimation of the robustness and resilience of a CP is performed by comparing the pdf associated with activity duration and the corresponding drag or TF values. For the activities that are part of the CP (see Figure 4), the activity duration pdf is compared with the duration point value used for the outage schedule (indicated in green in Figure 4). The portion of the pdf greater than the employed duration point value automatically adds delay to the CP. The portion of the pdf lower than the employed duration point value indicates the possibility that plant resources (i.e., crew personnel) can become available to reduce completion time of parallel or subsequent activities (see CP resilience).

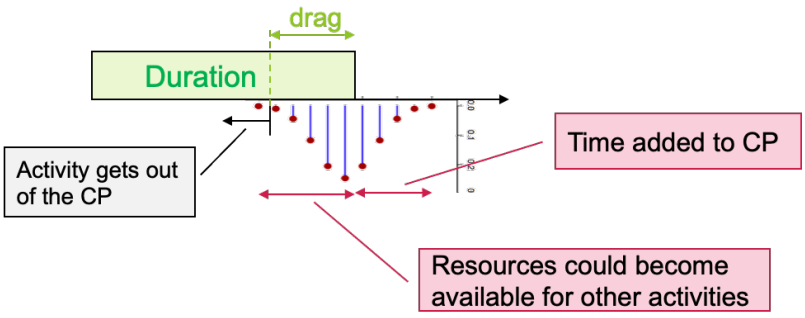

Figure 4. Analysis of activity duration uncertainty (represented through a histogram shown in blue) for an activity which is part of CP.

Similar discussion applies for activities not on the CP (see Figure 5): the portion of the pdf lower than the employed duration point value would allow plant resources to be available to reduce completion time of other activities. The portion of the pdf greater than the TF implies that such activity would become part of the CP. In this situation, the CP would change which would negatively impact outage completion time. The remaining portion of the pdf (i.e., located within the TF) is characterized by the fact that such activity duration delay would not affect the CP (see CP robustness).

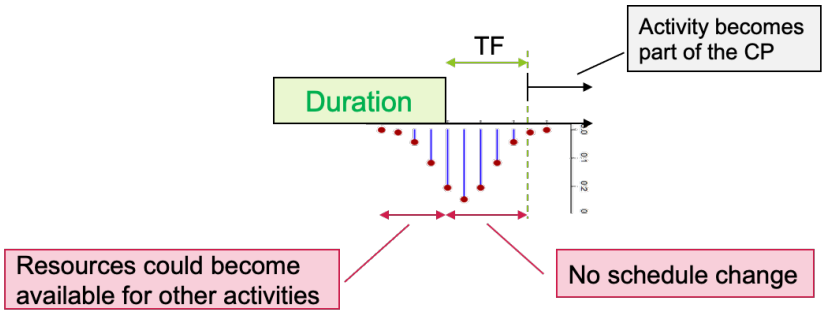

Figure 5. Analysis of activity duration uncertainty (represented through a histogram shown in blue) for an activity which outside the CP.

The development of analytical tools designed to assess CP uncertainty, robustness, and resilience started during fiscal year 2023 (FY-23). Such tools are based on the Idaho National Laboratory (INL)-developed open-source code RAVEN[2]. Here, the user has the possibility to import the outage schedule into RAVEN through an input file. As an example, the input file for the outage schedule based on the plan shown in Figure 1 is described in Figure 6: each of the eight activities are initialized with their specific duration time while the outage plan structure is provided as a graph structure where, for each node of the graph (i.e., an activity), the directly dependent activities are defined. Provided an outage schedule (e.g., the one shown in Figure 6), the model is able to determine the CP and the scheduled completion time.

Using RAVEN, it is possible to propagate uncertainties associated with activity duration through the outage schedule. This can be performed by defining a pdf for each activity duration and choosing the desired sampling strategy (e.g., Monte-Carlo or Latin Hypercube Sampling). Once the sampling has been completed, RAVEN provides a database that contains a pdf of the CP completion time and an alternative set of CPs.

[2] Official website: https://raven.inl.gov/SitePages/Overview.aspx

```
class project():
  start = Activity("start", 10)
  b     = Activity("b",     20)
  c     = Activity("c",      5)
  d     = Activity("d",     10)
  f     = Activity("f",     15)
  g     = Activity("g",      5)
  h     = Activity("h",     15)
  end   = Activity("end",   20)

  graph = {start: [f,b,h],
           b    : [c],
           c    : [g,d],
           d    : [end],
           f    : [g],
           g    : [end],
           h    : [end],
           end  : []}
```

Figure 6. Outage plan (also shown in Figure 1) defined in our outage analysis tools.

As an example, the pdfs associated with each activity of the example outage plan shown in Figure 1 are listed in Table 1. The duration uncertainties were propagated using RAVEN. The results are shown in Figure 7 which presents the pdf of the CP completion time (left plot), and the histogram of the possible CPs. Of interest is that the original CP (i.e., A-B-C-D-E) occurs with a 0.762 likelihood while the CP indicated as A-B-C-G-E occurs with a 0.237% likelihood. Last, the CP A-F-G-E occurs with a 2.E-4 likelihood.

Table 1. Set of pdfs associated with each activity of the outage plan shown in Figure 1.

Activity ID	pdf
Start	U(8,12)
B	U(15,25)
C	U(4,8)
D	U(8,12)
End	U(19,25)
F	U(12,20)
G	U(4,12)
H	U(12,21)

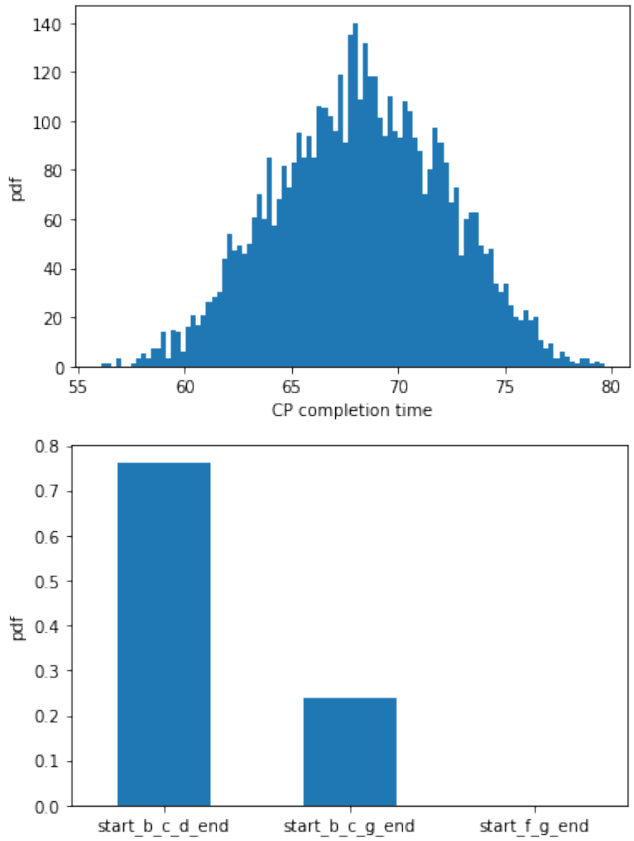

Figure 7. Propagation of the uncertainties specified in Table 1 for the outage shown in Figure 1: CP completion time (left plot), and the histogram of the obtained CPs.

The next step is the assessment of CP resilience and robustness. CP robustness Rob can be computed by analyzing the fraction of the samples characterized by a CP completion time less than the base CP completion time. More precisely, provided the pdf of CP completion time pdf^{CP_time}, CP robustness Rob can be solved analytically by integrating it up to base CP completion time (indicated as CP_time_base) as follows:

$$Rob = \int_{-\infty}^{CP\_time\_base} pdf^{CP\_time}(t)\, dt \tag{1}$$

For the case shown in Figure 1, the base $CP\_time\_base = 65$. The concept of resilience will be further developed in FY-24 since considerations of resources must be included.

3. MINING ACTIVITY DURATION DATA

3.1 Use-Cases

In this effort, two datasets were used from nuclear facilities. Both datasets contained a brief activity description, usually in the order of less than ten words, and timestamps of when the activity was predicted to start/end, and when it actually started/ended. Each activity had an alphanumeric code that seemed to follow a certain undefined structure. Given the sparsity of the activity description text, it was desired to connect each activity to the actual work scope performed. However, there was no connection established between an activity and the corresponding work order, and the outage work management data were not available. This was the case for both datasets. This section describes the performed evaluation determining whether it is possible to predict activity durations from the short activity description. This evaluation was conducted using two approaches: (1) a natural language processing approach and (2) a semantic text-mining approach.

3.2 Natural Language Processing for Duration Prediction

Using the provided dataset, this section describes exploration of a direct correlation between word occurrence in the activity description and the activity's predicted duration. This is followed by establishing the correlation between word occurrence in the activity description and the activity actual duration.

To examine how well the scheduler is predicting the activity durations, a plot is generated to correlate actual times needed to complete the job versus the predicted times in the schedule. Three different correlation methods were used as shown in Table 2 and each method was compared to a random data correlation. Pearson's r [1] is a normalized covariance between the planned hours and the actual hours taken to complete the activity. If the predictions and actual hours are centered (mean of zero), covariance is the prediction multiplied by the actual hours for each datapoint. The average of those products is covariance. To normalize covariance, it is the divided by both the standard deviation of the predictions and the actual hours. This gives Pearson's r. The correlation coefficient found using Pearson's r coefficient is 72.8%. Details on the other two methods can be found in [2,3]. Spearman's Rho is based on the monotonic correlation of the two variables (i.e., if one increases, the other increases as well). Kendall's Tau is similar but differs in the way the monotonic behavior is mathematically described.

Table 2. Summary of the correlation results using three different correlation methods.

Correlation Type	Original Forecast	Machine Learning Forecast
Log10 Transformed Data Pearson's r	72.8%	77.1%
Spearman's Rho (transformed or untransformed)	67.7%	73.5%
Kendall's Tau (transformed or untransformed)	55.5%	57.5%
Random Data (r, Rho or Tau)	0%	0%

The data are also shown in Figure 8. The number of datapoints used for this evaluation was approximately 1,000. The red error margins represent the margin for where 95% of the data are located and is calculated as[3]: red upper/lower margin = mean actual value ± 1.96 of the error standard deviation.

[3] The margins are created after the errors are smoothed out using a 3rd to 5th order polynomial of red error margin as a function of the fitted values.

The 95% uncertainty is on the order of tens to more than a hundred of hours. This is especially visible for the activities that are around the typical mean time of an activity. From this analysis, we can conclude that either the scheduler is unable to accurately predict the activity duration or that the plant staff are not logging the actual completion times, causing this very large discrepancy.

The hypothesis tested here is whether a machine can perform better in predicting activity durations. A machine learning model was developed using a fine-tuned Sentence-BERT (SBERT) transformer [4] on the activity description using a masked language model. A masked language model has the task of deleting words from a task description in the input to the neural network and then trying to predict the correct words that were deleted. This is a type of autoencoder and it is therefore unsupervised.

Next, the supervised task of predicting activity durations using a regressor is performed. The transformer neural network fine-tuned previously would output 7,168 dimensional embeddings. These embeddings were used to train a CatBoost regressor [5] on 80% of embeddings, then the CatBoost model forecasted 20% of the embeddings. The number of datapoints used for this evaluation was also approximately 1,000. All activities with fewer than 0.25 hours worked were discarded. The results of the prediction are shown in Table 2. The data are shown in Figure 9. The correlation coefficient is 0.77, indicating more consistent results. However, given the log scale use, this also represents tens to more than a hundred hours of uncertainty. Therefore, it is apparent that a machine cannot predict the activity duration given the short activity description.

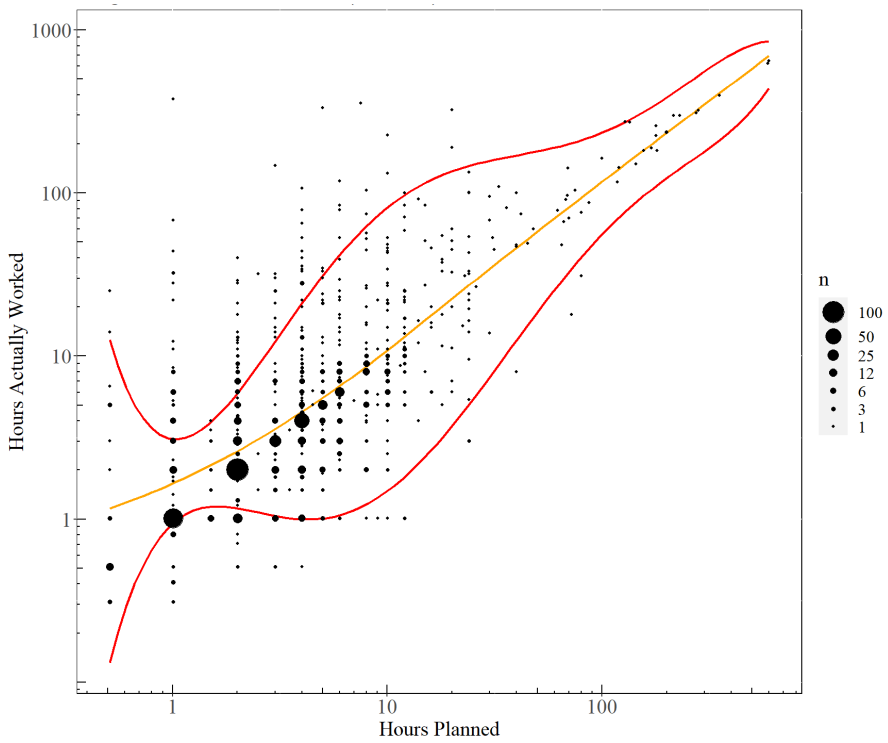

Figure 8. Plot of work hours forecasted by the scheduler versus the actual hours worked.

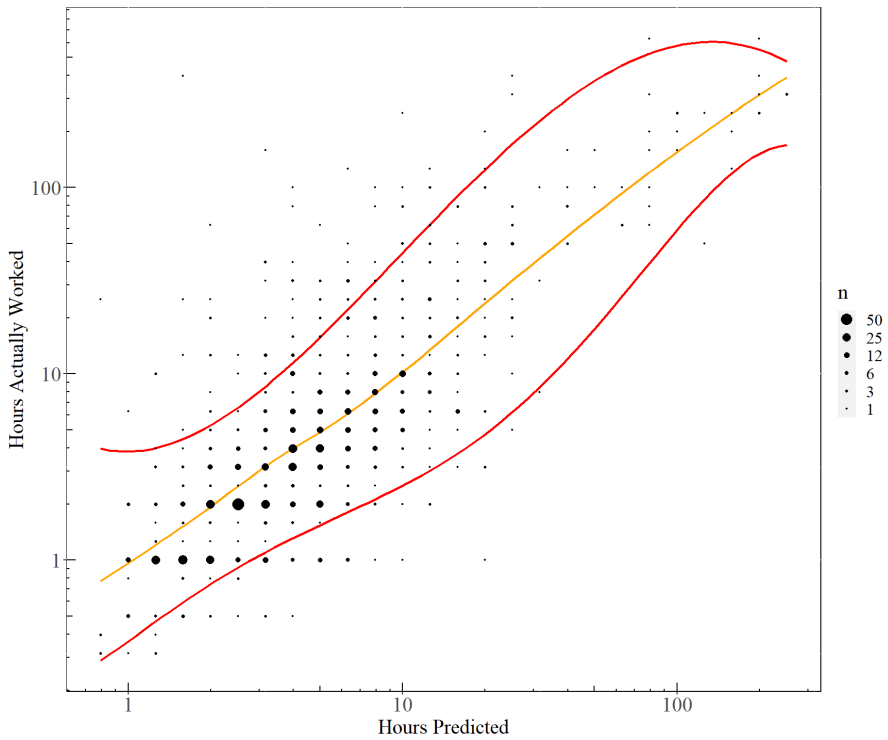

Figure 9. Plot of work hours forecasted by machine learning versus the actual hours worked.

To further compare the performance of the scheduler versus the machine, the differences between scheduler-predicted planned and actual activity durations are plotted against the difference of the machine-predicted durations. This is only performed for 20% of the data since the remaining 80% of the data were used for training of the CatBoost + SBERT transformer algorithm. The result is shown in Figure 10. The error statistics of both the planned/scheduled and predicted durations show that a machine tends to predict higher values, while the scheduler tends to predict smaller values. The figure also shows that the machine predictions resulted in smoother prediction since it was able to predict fractions of hours, unlike the scheduler, often in 0.25–1 hour increments. To overcome this distinction, a classifier is used instead of a regressor to place each activity into the bins shown in Table 3. For each bin, a count of the misclassification is listed in each cell. The table indicates that the machine was better in predicting low number of hours but suffered as the number of hours increased in comparison to the scheduler.

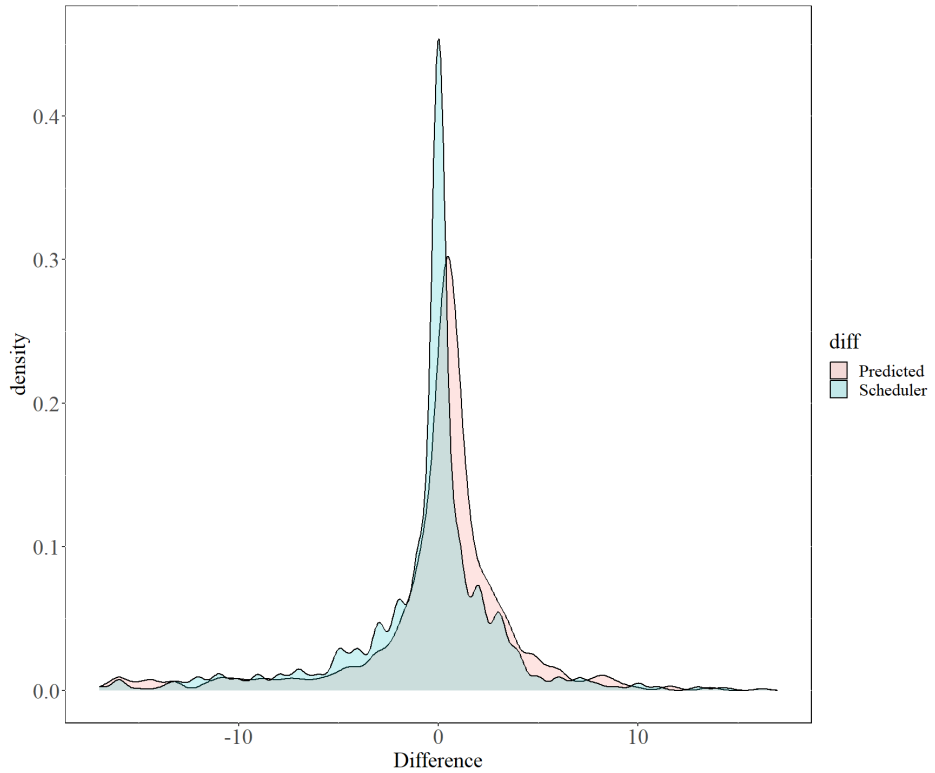

Figure 10. Plot of difference of work hours forecasted by machine learning or scheduler and the actual hours worked.

Table 3. Confusion matrix discretized for 1, 3, 5, 10, 100, 1,000, and more hour differences.

Error Type	(0,1)	(1,3)	(3,5)	(5,10)	(10,100)	(100,1000)	1000+
Planned Error	534	207	89	72	150	13	0
Machine-Predicted Error	428	293	100	73	149	22	0

3.3 Text Semantic Similarity for Duration Prediction

A parallel research direction to the one shown in Section 3.2 was to evaluate completion time of an activity using text semantic similarity. The basic idea is to identify the subset of activities performed in previous outages that are similar to the activity being queried. Then the temporal distribution of the queried activity can be determined by collecting the historical completion time from the subset of past activities.

An example of textual similarity is shown in Figure 11 where two activities with similar semantic meanings are compared which brings up the importance of data cleaning and data curation. The example provided in Figure 11 suggests that if we were to perform a simple word-to-word similarity between those two activities, they would be very dissimilar. On the other hand, if the historical activity were to be cleaned (e.g., through spell checking, and abbreviation identification and expansion), then it would be transformed into "[ACC01-B] PRESSURE TRANSMITTER CALIBRATION." Consequently, the two activities would be very similar.

The elements required for the semantic similarity analysis are:

- The set of past outage activities. This set might be partitioned on several datasets, a dataset for each outage. Outage of different plant units, different plants, or different utilities can be gathered to improve analysis results.
- A computational method designed to compute the semantic similarity between two activities (i.e., the queried and historic activity) which would generate a point value which measures "how similar" the two activities are. An important note here is that the computational time for such a method needs to be very small since the similarity search for a queried activity in a database of tens of thousands of past activities needs to be performed within minutes.

In this project, we focused on the development of the semantic similarity method and on the testing of this method on several outage databases. The following sections provide details of the development and present a high-level overview (to mask proprietary data) of the obtained results.

Figure 11. Example of semantic similarity between a queried and a historical outage activity.

3.3.1 Textual Similarity Analysis

Words, sentences, and documents similarity analyses are an active part of recent NLP methods development, and these analyses play a crucial role in text analytics such as text summarization/representation, text categorization, and knowledge discovery. There is a wide variety of methodologies that have been proposed during the past two decades. Mainly, these techniques can be classified into five groups: (1) lexical knowledge base approach, (2) statistical corpus approach (word co-occurrence), (3) machine learning/deep learning approach, (4) sentence structure-based approach, (5) hybrid approach. However, there are several common major drawbacks for these approaches: (1) computationally inefficient, (2) lack of automation, (3) lack of adaptability and flexibility. In this research, we are trying to address these drawbacks by developing a tool that can be used generally in any application requiring a similarity analysis.

As shown in Figure 12, we are trying to leverage part of speech (POS), disambiguation, lexical database, domain corpus, word embedding/vector similarity, sentence word order, and sentence semantic analysis to calculate sentence similarity. POS is used to parse a sentence and tag each word/token with POS tag and syntactic dependency (DEP) tag. This information provides syntactic structure information (i.e., negation, conjecture, and syntactic dependency) about the sentence that can be used to guide the similarity measuring process. The disambiguation approach is employed to determine the best sense of the word, especially when coupled with a specific domain corpus. It will ensure the right meaning of the words (e.g., the right word synsets in the lexical database) among the sentence for comparison.

Then, a predefined word hierarchy from lexical database (i.e., WordNet) is used to compute the word similarity. However, some words are not contained in the lexical database since it only connects four types of POS – nouns, verbs, adjectives, and adverbs. Moreover, these words are grouped separately and do not have interconnections. For instance, nouns and verbs are not interlinked (i.e., the similarity score between "calibration" and "calibrate" is 0.091 when using WordNet). In this case, machine-learning-based word embedding is introduced to enhance the similarity calculation. For the "calibration" and "calibrate" example, the similarity score becomes 0.715 instead. The next step is to compute sentence similarity by leveraging both sentence semantic information and syntactic structure. The semantic vectors are constructed using the previously introduced word similarity approach, while the syntactic similarity is measured by word order similarity. The following sections further describe each of the steps in more details.

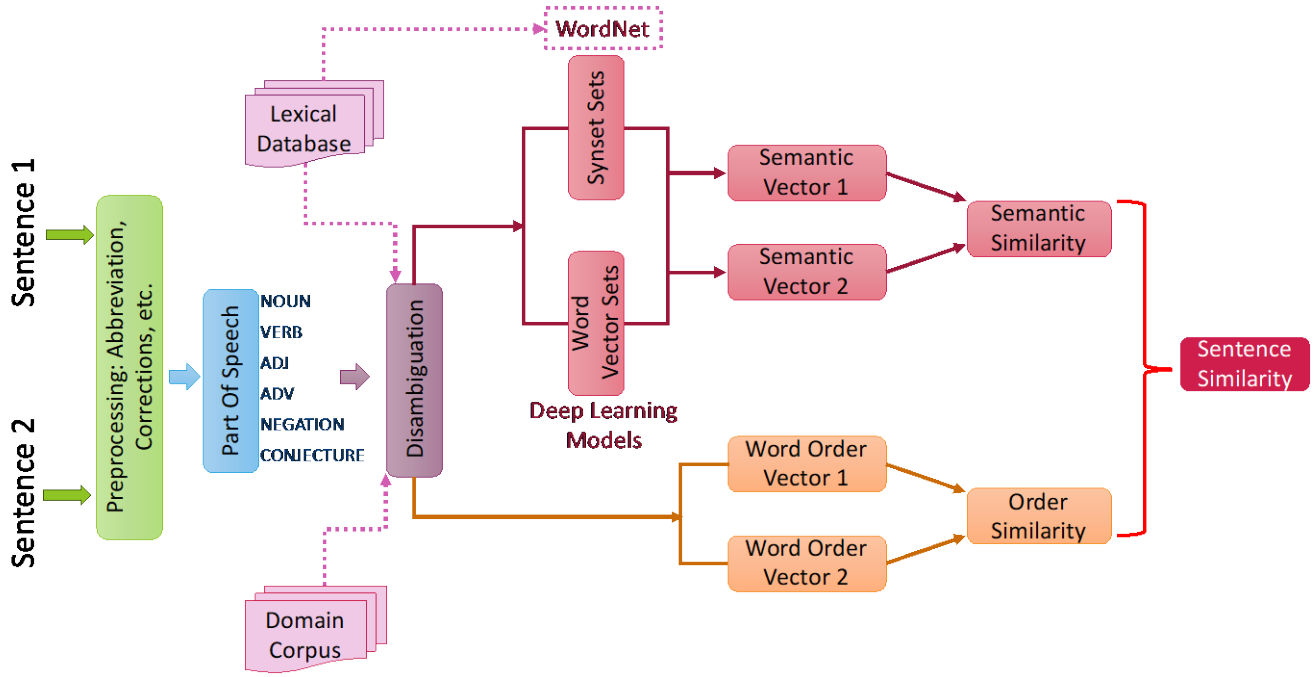

Figure 12. Illustration of sentence similarity calculation.

3.3.2 Part of Speech (POS) for Similarity Analysis

POS provides information about word types and morphological features, and dependency parsing provides dependency syntactic information between words. Utilizing POS and dependency parsing can help to identify the important information, such as NOUN, VERB, ADJ, ADV, negation, conjecture, subject, and object, which will be used to compute the sentence syntactic similarity.

3.3.3 Lexical Database

Lexical databases, such as WordNet, have semantic connections between words which can be utilized to determine the semantic similarity of the words. WordNet is a lexical information database originally created by Princeton University. It contains words, their meanings (e.g., synsets), and their semantic relationships which are stored in a hierarchy tree-like structure via linked synsets. Each synset denotes the precise meaning of a particular word, and its relative location to other synsets can be used to calculate the similarity between them.

As summarized in Reference [6], there are many different methods to compute word similarity using WordNet and sometimes these methods are combined to enhance the similarity calculation. In this work, we employ the method proposed by [7] to compute the similarity score between two words/synsets as presented in Eq. (2). This method combines the shortest path distance between synsets and the depth of their subsumer (e.g., the relative root node of the compared synsets) in the hierarchy. In other words, the similarity score is higher when the synsets are close to each other in the hierarchy, or their subsumer locates at the lower layer of the hierarchy. This is because the lower layer has more specific features and semantic information, as compared to the upper layer.

$$S_w(w_1, w_2) = f_{length}(l) \cdot g_{depth}(d) = e^{-\alpha l} \cdot \frac{e^{\beta d} - e^{-\beta d}}{e^{\beta d} + e^{-\beta d}} \quad (2)$$

where $\alpha \in [0,1], \beta \in [0,1]$ are parameters scaling the contribution of shortest path length and depth respectively.

The optimal values of α and β are dependent on the knowledge base used and can be determined using a set of word pairs with human similarity ratings. For WordNet, the optimal parameters for the proposed measure are: $\alpha = 0.2$ and $\beta = 0.45$, as reported in Reference [8].

3.3.4 Associating Word with the Best Sense (Disambiguation and Domain-Specific Corpus)

A sense represents the precise meaning of given word under specific context. Disambiguation is the process to identify the best sense for a word in the context of a statement. Without proper disambiguation, errors could be introduced at the early stage of similarity calculation when using lexical databases. For example, in WordNet synsets are used to denote the senses of the word and they are linked to each other by their explicit semantic relationships. When different synsets are used in calculating word pair similarity, their semantic relationship can be drastically different, which can significantly affect the similarity score. In this work, we try to disambiguate the word sense by considering the context of the word. One way to do this is to take into account the surrounding words since they can provide the contextual information. However, this may not work for simple or short sentences. In this case, the domain-specific corpus can be leveraged to disambiguate the word. Once the best senses are identified for the words, the word similarity measure from Section 3.3.3 can be employed.

3.3.5 Word Embedding/Vector

A word embedding or word vector is typically a numerical vectorization of words or documents. It maps words with semantic similarities to have close embedding vectors. Thus, word embedding can be used to measure semantic similarities utilizing cosine similarity metrics between the embedded vectors. This is especially useful when WordNet fails in situations such as similarities between words that have different POS tags. In this work, word embedding is leveraged to assist the word similarity calculation. Once the similarity score from the WordNet similarity calculation is below 0.2 (e.g., the two words are not similar), the word embedding similarity calculation is employed.

3.3.6 Sentence Similarity

As proposed in Reference [7], sentence similarity contains semantic and syntactic similarity. Semantic similarity is captured via word semantic similarity as discussed in previous sections, while syntactic similarity is measured by word order similarity. Word order similarity is a way to assess sentence similarity considering order of words. As described in Reference [7], the semantic vectors and word order vectors are constructed and can be used to compute the sentence similarity. Here, we briefly introduce the methods to construct these vectors and refer the reader to Reference [7] for more details.

Given two sentences, T_1 and T_2, a joint word set is formed (e.g., $T = T_1 \cup T_2$) with all the distinct words from T_1 and T_2. The vectors derived from computing word similarities in $(T, T1)$ and $(T, T2)$ are called the semantic vectors, denoted by s_1 and s_2, respectively. Each entry of the semantic vectors corresponds to the maximum similarity score between a word in T and a word in T_1 or T_2, so the dimension equals the number of words in the joint word set. The semantic similarity between two sentences is defined as the cosine coefficient between two vectors:

$$S_s = \frac{s_1 \cdot s_2}{\|s_1\| \|s_2\|} \tag{3}$$

As proposed by Reference [7], the word order similarity of two sentences is defined as:

$$S_r = 1 - \frac{\|r_1 - r_2\|}{\|r_1 + r_2\|} \tag{4}$$

where the word order vectors r_1 and r_2 are formed from $(T, T1)$ and $(T, T2)$, respectively.

For example, for each word w_i in T, the r_1 vector with the same length of T_1 is formed as follows: if the same word is present in T_1, the word index in T_1 is used as the value for r_1. Otherwise, the index of the most similar word in T_1 will be used in r_1. A preset threshold (i.e., 0.4) can also be used to remove spurious word similarities. In this case, the entry of w_i in r_1 is 0.

Semantic and syntactic information (in terms of word order) each play a role in measuring the similarity of sentences. Thus, the overall sentence similarity is defined in [7] as a combination of:

$$S(T_1, T_2) = \delta S_s + (1 - \delta) S_r \tag{5}$$

where $\delta \in [0, 1]$ represents the relative contribution of semantic information to the overall similarity computation.

3.3.7 Application of Similarity-Based Methods to Outage Analysis

Section 3.1 provided an initial overview on how methods based on the textual semantic similarity can be employed during the planning phase of an outage to assess completion time variability of specific activities. There, a note was made about the importance of data curation (e.g., cleaning and reconstruction) of the textual elements that describe each activity. We highlight here again how our developed methods balance word and semantic sentence similarities. Thus, a sub-optimal data curation analysis might negatively impact the search for similar activities.

More specifically, the process of data curation performed for all historical outage activities includes the following steps:

1. *Removal of component IDs*. The presence of specific asset or system IDs (e.g., accumulator ACC-01B in Figure 11) do not necessarily provide any type of information from a semantic point of view and, hence, they can be removed from the actual text. This can be accomplished by either parsing the activity text or providing a list containing the full list of plant asset or system IDs. During our testing (see Section 3.3.8) such a list was not available, and we relied on an empirical method designed to remove all words containing a mixture of characters, numbers, and symbols.
2. *Abbreviation handling*. NPP outage activities are usually short sentences which often contain abbreviations. The presence of abbreviations negatively impacts the ability to extract knowledge from such texts. Hence, we have developed an NLP pipeline designed to identify abbreviations and replace them with their corresponding complete words. The starting point is a library of abbreviations that have been collected from documents available online. This library is basically a dictionary which relates an identified abbreviation to the corresponding set of words. A challenge here is that a single abbreviation might have multiple words associated with it. Similarly, a word might have multiple ways to be reduced. Handling of abbreviations in each sentence is performed by first identifying misspelled words. Then each misspelled word is searched in the developed library. If an abbreviation in the library matches the misspelled word, then it is replaced by the corresponding complete word. If no abbreviation in the library is found, then we proceed by searching for the closest one. If multiple words match the obtained abbreviation, then the word that fits most of the sentence context is chosen.
3. *Spellcheck*. After the abbreviation handler method is completed, the remaining misspelled words are parsed through our spellchecking methods for a last correction.

Once historical plant outage data have been cleaned, the similarity value between the queried activity and each historical activity is determined. This results in an array of similarity values with dimensionality identical to the number of historical activities, and the corresponding array (with identical dimensionality) containing the activity durations.

The computation of the predicted duration of the queried activity is determined by considering both the similarity and the duration arrays. More precisely, by setting a similarity threshold sim_{thr} (typically in the 0.7–0.9 range[4]), we are collecting elements of the duration array so the corresponding similarity measure is greater than sim_{thr}. A relevant note to be highlighted here is that if the queried activity has never been completed in past outages, then no similar past activities with similarity value above sim_{thr} will be found. This approach does not, in fact, perform any type of regression.

3.3.8 Analysis Example

An initial application of the developed methods was performed using the dataset provided by an existing United States (U.S.) NPP. This dataset contains activities performed during five outages. The number of activities varies from outage to outage but the data indicate activities in the [12000, 14000] range. The data cleaning described in Section 3.3.7 was performed for the activities contained in each of the five outages. This process required a large computational time, about 4 hours for each outage using an off-the-shelf laptop, which can be easily reduced by parallelizing such computation[5]. However, note that such computation is performed only once (i.e., when new outage data are available).

A relevant feature of the provided datasets is that some activities are categorized using plant-specific labels. A label indicates the type of work performed in an activity (e.g., electrical, chemical, instrumentation and control). Note that a good portion of outage activities (about 30%) is not labeled. For those activities, the label NaN was assigned. About a hundred unique labels were identified. Figure 13 shows the histogram of the number of activities contained in each of the 98 identified labels (including NaN) for a single outage data set. The x-axis that lists the 98 labels is masked to remove plant proprietary data. Such activity labeling can be used to limit the number of historic activities searched for. More specifically, if a label can be assigned to the queried activity, then the similarity search can be performed only for those historic activities that have the same label.

Finally, regarding the similarity search, we performed several tests:

1. Choose one outage dataset out of the available five datasets, and consider the other four datasets as the testing dataset.
2. Randomly sampled an activity from the dataset chosen in 1.
3. Determined the similarity values between the activity sampled in 2 and all the activities contained in the testing dataset. Figure 14 shows a snapshot of the actual code and the obtained results. Here, the similarity method (indicated as SentenceSimilarityWithDisambiguation) is used to measure the similarity value (indicated as similarity) between a queried activity (indicated as queryAct) and all activities contained in the instance of a cleaned outage data (indicated as cleanData2). The results are masked to hide proprietary information, but Figure 14 shows the obtained similarity values. For the shown case, about 20 similar activities were found.

[4] Recall that a similarity measure is in the (0,1] range where perfect similarity is indicated with the unitary value, while very low similarity values (near 0) will be assigned for dissimilar textual elements.

[5] The INL-developed code RAVEN (https://github.com/idaholab/raven) can be employed to parallelize the computation.

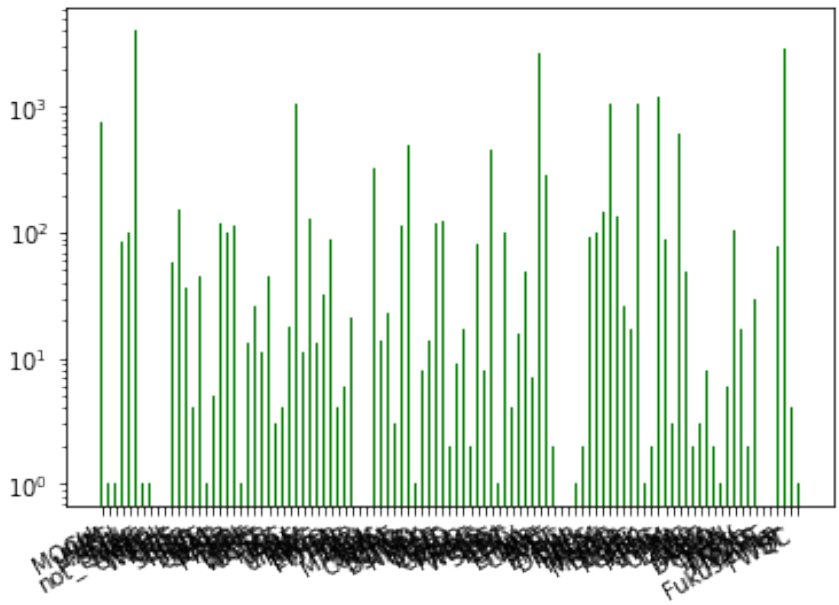

Figure 13. Histogram of the number of activities contained in each of the 98 identified labels (including NaN) for a single outage data set.

```
for index,sent in enumerate(cleanData2):
  similarity = simUtils.sentenceSimilarityWithDisambiguation(queryAct, sent, senseMethod='simple_lesk', simMethod='path', delta=1.0)
  print(str('{:.3f}'.format(similarity)) + " - " + sent)
```

```
0.939 -
0.089 -
0.642 -
0.642 -
0.283 -
0.356 -
0.511 -
0.221 -
0.261 -
0.449 -
1.000 -
0.707 -
0.577 -
0.734 -
```

Figure 14. List of obtained similar activities (field on the right which has been masked to preserve proprietary data) with the corresponding similarity value (on the left).

Last, we employed the developed similarity methods to analyze the degree of similarities between all the activities that belong to the same outage dataset.

1. Determined the similarity matrix $M = [s_{i,j}]$ (i.e., a squared symmetric matrix of size equal to the number of activities contained in the outage dataset) where each element $s_{i,j}$ of this matrix contains the similarity value between activity i and j of the outage dataset.
2. Determined the equivalent distance matrix $D = [d_{i,j}]$ (which is also a squared symmetric matrix with a size identical to M) where each element $d_{i,j}$ contains the distance between activity i and j of the outage dataset; in our case $d_{i,j}$ has been calculated simply as $d_{i,j} = 1.0 - s_{i,j}$.
3. Performed hierarchical clustering on the obtained matrix D. The outcome of this analysis is a dendrogram (see Figure 15) which is an effective way to visualize the relative distance of a set of data points for different resolution levels. The dendrogram shows that only few patterns can be identified by looking at dendrogram branches that are vertically spaced (e.g., right portion of the green portion of the dendrogram).

As a final note, we are developing an approach to evaluate and quantity the performance of the developed methods. These methods will be finalized in FY-24.

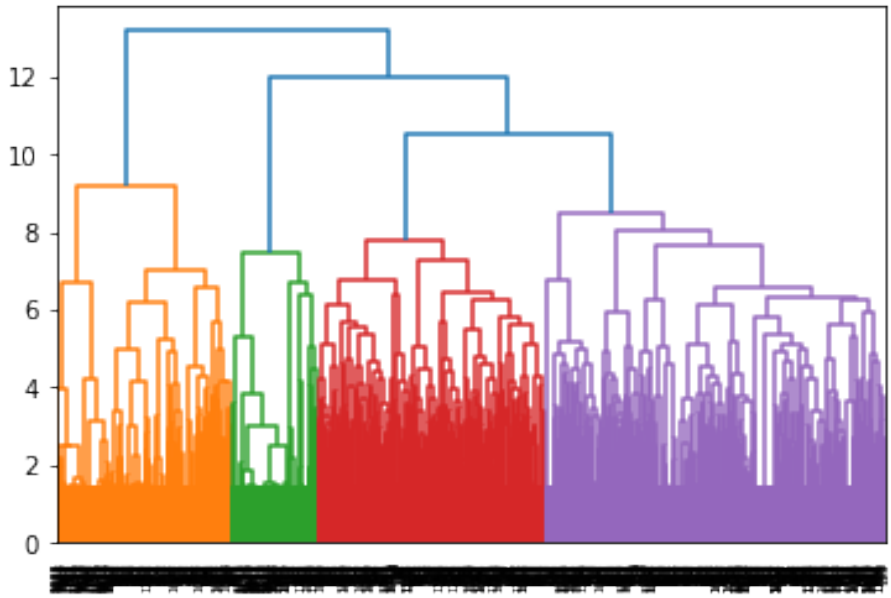

Figure 15. Dendrogram obtained from the similarity values of the activities contained in a single outage dataset.

4. CONCLUSION AND FUTURE VISION

In summary, the objective of this research project is to develop methods and tools to help plant staff create an outage schedule with a high probability of completion in the desired timeframe. Many plants continue to struggle with completing outages within the planned duration. Most utilities use the CP methodology to analyze and optimize schedules. To improve the current CP approach used by most utilities, we are investigating a method to calculate schedule resilience by creating a model of the schedule. A resilient schedule is one that is analyzed and adapted to find task duration uncertainties that can be reorganized to absorb completion time variability. This tool would provide the staff with margins for highly uncertain, non-CP activities. A resilient schedule is also one that has the capability to absorb the expected amount of scope growth without significant disruption to the planned duration.

The tools supporting outage activities may be used during outage planning and during outage progression. The outage planning can be improved by identifying activities where a planned duration is significantly different compared to the historical actual duration for the same activity. This would help to make the outage schedule more realistic. The planning tool would also identify non-CP strings of activities that have high uncertainties in their durations and therefore have a high probability of becoming a CP. Being informed about the high-risk activities strings allows alternative schedule planning options, which improves schedule resilience. The tool could highlight highly uncertain strings of activities that must remain in the schedule so that outage staff can maintain proper focus and oversight on those specific tasks to improve the chances of on-time completion. During the outage execution phase, the proposed tool would make recommendations for the best schedule for emergent work activities to minimize the chances or the magnitude of outage extensions.

In the initial stages of this project, various ML/AI methods were investigated to assess the possibility of automatically assigning activity duration distributions based on historical outage performance data analysis. Facilities provided schedule data and several methods were used to evaluate the schedule duration and variability. In general, it was determined that many ML/AI techniques fall short in interpreting the activity descriptions assigned by the utilities. In many cases, the use of abbreviations for activity descriptions limited the ability of the ML/AI tools to match an activity to other similar activities in the data set. The investigation of ML/AI capabilities will continue in the next phase of the project.

We also investigated using ranges of values described by a probability distribution function to represent an activity duration instead of a single value duration. This work will also be expanded in the next phase of the project. In cases where data are not sufficient to determine a duration distribution, a schema will be developed to assign a standard duration distribution based on generic average completion time for common types of work activities such as valve refurbishment, erecting scaffolding, circuit breaker refurbishment, etc.

While the concept of schedule resilience is understandable, metrics need to be developed based on schedule modeling for automated optimization and recommendations. These metrics for schedule resilience will assist the outage schedulers in visualizing potential issues and provide useful information for schedulers to evaluate alternative scheduling options. Initial concepts for resilience metrics are presented in this report and will be refined and expanded as the research continues. Additional future work will be done to integrate information available in the work management databases and improve the assignment of distributions to the activity durations. The project team will also create example use-cases using representative example schedules to develop user interfaces and demonstrate concepts for schedule resilience presentation and recommendations for optimization.

5. REFERENCES

1. Bollen, K.A. and K. H. Barb. 1981. "Pearson's r and coarsely categorized measures." *American Sociological Review* 46(2): 232-239. http://dx.doi.org/10.2307/2094981.
2. Sen, P.K., 1968. "Estimates of the Regression Coefficient Based on Kendall's Tau." *Journal of the American Statistical Association, 63*(324): 1379-1389. https://doi.org/10.1080/01621459.1968.10480934.
3. Daniel, W.W. 2000. "Applied Nonparametric Statistics." Cengage Learning; 2nd edition, ISBN 10-0534919766, ISBN-139780534919764.
4. Reimers, N., and I. Gurevych. 2019. "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks." In proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. https://doi.org/10.18653/v1%2FD19-1410.
5. Hancock, J. T., and T. M. Khoshgoftaar. 2020. "CatBoost for big data: an interdisciplinary review." *Journal of Big Data*, 7(1): 94. https://doi.org/10.1186/s40537-020-00369-8.
6. Navigli, R., and F. Martelli. 2019. "An overview of word and sense similarity." *Natural Language Engineering* 25(6): 693–714. http://dx.doi.org/10.1017/S1351324919000305.
7. Li, Yuhua, et al. 2006. "Sentence similarity based on semantic nets and corpus statistics." *IEEE Transactions on Knowledge and Data Engineering* 18(8): 1138–1150. http://dx.doi.org/10.1109/TKDE.2006.130.
8. Li, Yuhua, Zuhair A. Bandar, and D. McLean. 2003. "An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources." *IEEE Transactions on Knowledge and Data Engineering* 15(4): 871–882. http://dx.doi.org/10.1109/TKDE.2003.1209005.