# Transforming metal production by maximizing revenue generation with Operational AI

Crick Waters[1], Beverly Klemme[1], Raj Talla[1], Prerna Jain[1], Nikunj Mehta[1]

[1]Falkonry Inc.
Sunnyvale, CA 94087 USA
Phone:+1 408 461 9286

Emails: {crick@falkonry.com, beverly.klemme@falkonry.com, raj.talla@falkonry.com, prerna.jain@falkonry.com, nikunj.mehta@falkonry.com}

ABSTRACT

Recently, Artificial Intelligence (AI) and Machine Learning (ML) techniques have been used to solve complex operations problems. However, scaling ML/AI across a multitude of equipment types and use cases, a variety of signals, and over time with changing operations remains a significant challenge. This paper discusses how Falkonry's Operational AI platform learns, detects, and predicts conditions in continuous casting. This methodology is scalable across use cases and time without the need for data scientists. Precedent detection of impending equipment failures allows operations to schedule necessary maintenance interventions, thus avoiding loss of production due to unexpected downtime events.

Keywords: Operational AI, Machine Learning, Time Series Data, Predictive Operational Excellence, Predictive Maintenance, Continuous Casting, Reliability, Condition-Based Maintenance

## INTRODUCTION

The advent of IoT sensors, edge devices, and connectivity standards in factories has pushed data collected from industrial automation projects into the cloud. This new data availability has spawned a revolution in both the underlying methods and scale of data analytics – creating smart factories whose performance and behaviors are managed by digital interpretation of data. Insights are no longer limited to human inspection and domain knowledge but are now acquired using digital technologies and practices. Machine learning methods have emerged onto the industrial landscape and are rapidly evolving to meet the demand for this next generation of digital insights.

Industries that have the maturity and foresight to ride the wave of new and smart factory innovation have tremendous potential to create a competitive advantage by boosting productivity and increasing revenues. For example, cost savings can come from predicting equipment failures, shifting the emphasis from *preventive* to *predictive* maintenance, freeing up otherwise lost production time for incremental revenue generation.

Despite the evolution of sensors, devices, cloud computing, and innovations in machine learning, many factors challenge the practical application of machine learning and artificial intelligence in operations at scale. Scaling across multiple equipment types and use cases is one such challenge. Traditional approaches by data scientists and use-specific software development simply cannot scale - even with “auto ML”[1], shortening the machine learning cycle for algorithm selection. Not only is the custom-ML development cycle a substantial impediment to scale, so too is the effect of time on ML models as underlying production equipment is upgraded, raw materials change, product mixes evolve, and new failure modalities emerge through long-term use. Any or all these changes render custom-crafted ML models moot, leaving the manufacturer without a critical continuous analytic tool.

This paper presents Falkonry’s Operational AI as one solution to the challenges of scaling machine learning and AI in manufacturing. This intelligence-first approach[2] [3] to creating machine learning applications is based on a consistent, repeatable, time series classification technique robust to incomplete and irregular data, engenders expert knowledge capture, and leads to real-time actions to positively affect production. Underlying challenges in the real-time application of Operational AI to production systems are discussed. This paper further presents a real-world case study and how Operational AI is applied in steel manufacturing at production scale to predict equipment failure to reduce unexpected downtime, thus increasing equipment availability for revenue generation. Results and advantages of Operational AI approach from application to casting operations are discussed, followed by conclusions.

## OPERATIONAL AI AND CHALLENGES

### OPERATIONAL AI

Operational AI is a software platform to enhance operational excellence by avoiding operational events that disrupt manufacturing and production operations. As such a broad set of problems need to be solved for increasing operational excellence, such as reducing downtime, reducing injuries, increasing throughput, enhancing capacity utilization, reducing cycle time, increasing yield, reducing work-in-progress inventory, lowering mean cost per unit, reducing and eliminating defects and lowering changeover time. For such operational excellence solutions, Operational AI needs to offer a broad set of process and asset-agnostic capabilities.

Operational AI functions by applying computational methods to operational data to discover and report behaviors that engage operational experts and solicit their know-how through accessible interfaces. Operational AI is easy-to-use in that front-line practitioners (such as manufacturing engineers, reliability engineers, process engineers, maintenance managers, and those with similar experience) can use themselves. Operational AI produces applications that predict undesirable system behavior and can be evolved without assistance from data engineers or data scientists. Operational AI emphasizes solutions that enable front-line subject matter experts to use the solution on their own. Results are achieved more rapidly, scale more quickly, and do so at a lower cost than traditional AI allows because the coordination and direct costs of data science specialists are not required. In this paper, the term "Operational AI" refers to Falkonry's Operational AI platform [4].

Various manufacturing sectors such as energy, chemical, pharmaceutical, and metals need Operational AI today [5] [6] [7] [8]. AI/ML techniques prove to be promising for a variety of manufacturing applications across the value chain. These applications accelerate decision-making, minimize unanticipated failures, and improve logistics and utilization of resources.

Figure 1 shows Falkonry's Predictive Operational Excellence framework.

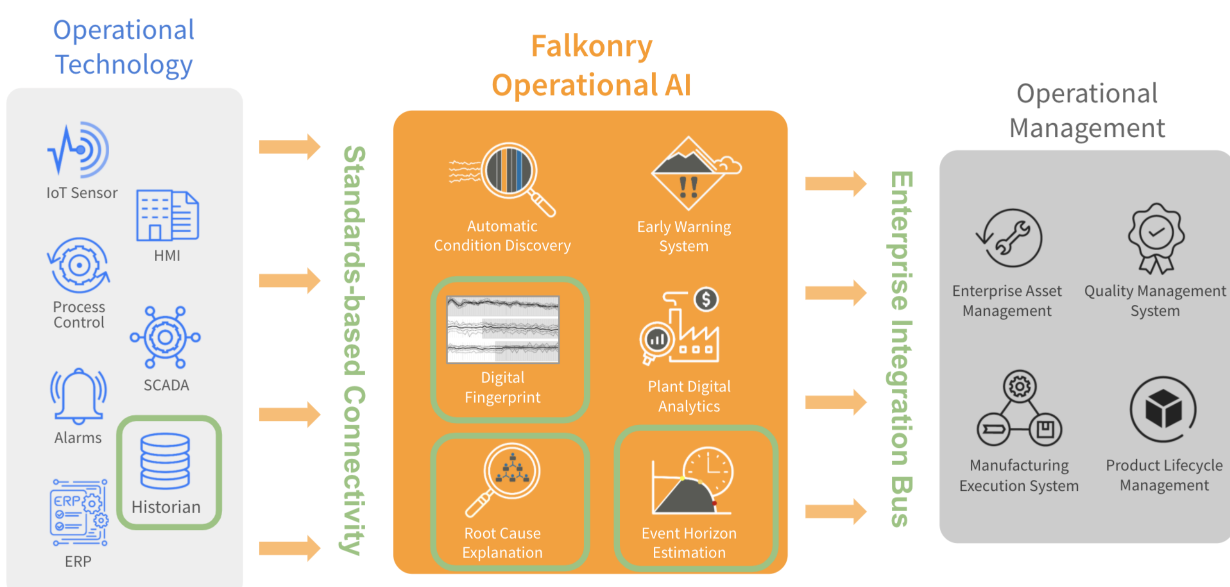


Figure 1: Falkonry's Predictive Operational Excellence Framework
*(the green boxes highlight application related to use cases described in this paper)

### Operational Technology Layer

Operational Technology consists of fine-grained, high sample rate data collected at the sensor and control level. Values such as Volts, Amperes, Hertz, bar, degrees, and position are sampled at rates as high as microseconds. These data originate from "operational technology,"[9] or "OT," sources that are part of the manufacturing systems. These OT sources include:

- SCADA (Supervisory Control and Data Acquisition),
- plant historians,
- IoT sensors
- HMI (Human-Machine Interface),
- PLCs (Programmable Logic Controller),
- DCS (Distributed Control System)
- process control systems, and
- alarm management systems

**Operational AI**

Operational AI performs data normalization, aggregation and analysis. Operational AI transduces the fine-grained data from OT[1] into actionable insights through discovery and detection of both known and novel operating conditions. Operational AI provides an explanation of detected conditions to assist with root cause analysis and an estimation of the remaining time to events of interest (such as consumable replacement or time to failure). Through practitioner interaction with digital twins and plant analytics, digital insights and recommendations are translated into decisions and converted to actions by sending workflow instructions to operational management systems.

**Operational Management**

Operational Management connects insights and recommendations from Operational AI to business operations and management processes, bringing intelligence to the overall operations facilitating business value objectives. Operational AI processes fine-grained OT data and turns it into inputs and recommendations for Operational Management systems.

The flow of information from OT sources through Operational AI to Operational Management systems maximizes value-creation across domains such as equipment maintenance, plant design, manufacturing management, energy management, and quality control while focusing on factors like costs, schedules, risks, and business priorities.

## CHALLENGES

Numerous challenges stand before the innovative steel manufacturer intent on increasing productivity without new capital investment using existing capital assets. Continuous casting of steel is a well-understood process. Once a caster is put into operation, the manufacturer can do little to change the physics of the caster to change rated production throughput. Rated throughput, however, is not continuously attainable in real-world applications.

Despite the maturity of continuous casting equipment and casting operations, valuable production time is lost to unscheduled downtime events in real-world applications. An "ordinary" continuous caster producing 150 tonnes per hour can generate $2,500,000 per day in production revenue. Conversely, a single day of lost production *loses* that equivalent $2.5M. Therefore, a manufacturer can release tremendous stored potential value by eliminating unscheduled production downtime.

Casting molten steel, not surprisingly, is hard on heavy equipment. Components wear under harsh conditions leading to failures or adverse product quality. However, the opportunity exists to apply machine learning techniques to detect early evidence of conditions leading to equipment failure. Early detection of such conditions is a warning to maintenance and production scheduling managers that downtime needs to be scheduled for repair before failures occur. Therefore, the first challenge is developing a machine learning application that can detect those early conditions that predict equipment failure and not being distracted by other operating modes that inevitably arise in industrial operations.

The next challenge is deploying ML applications into a production environment. Sensor and machine parameters are often available in SCADA, PLC, or from a data acquisition system such as IBA[10]. An Operational AI platform uses such operational data to create Operational AI applications that predict conditions observed in that data and these systems contain data that is not systematically normalized for machine learning. Operational AI platforms may use a small amount of recent data or large amounts of historical data to create applications for specific failure or operational modes that indicate pre-failure conditions. Operations practitioners need to be able to deploy those applications into continuous operation

---

[1] Sometimes data used by Operational AI may not originate in OT, such as raw material characteristics and production plan.

without costly and time-consuming software development and evolve them quickly over long periods of usage. The Operational AI platform must fit within the steel manufacturers' overall data and operational management software architecture. It must receive raw data while also providing insights to manufacturing management systems such as asset performance management (APM) or computerized maintenance management systems (CMMS).

Creating and deploying ML/AI applications for detecting and predicting equipment maintenance requirements is further complicated by two main issues: 1) relatively infrequent actual problems and 2) changes to operations over time. Industrial operations insist on maintaining low variance and long MTBF. As a result, problem conditions tend to be rare and unique - meaning that what one believes today to be a "complete" design will invariably miss a future, previously unknown, failure mode. Over the ordinary course of plant operations, maintenance will be performed, and operating conditions will change. Replacement parts may have different characteristics than original parts. Operating modes will evolve, and products will change as raw material sources shift to accommodate market, demand, and product dynamics. Furthermore, both of these challenges mean that ML/AI approaches must be self-regulating for change and agile to execute required changes. This requires Operational AI to be capable of discovering conditions on its own for being managed by operational experts. This characteristic is one of the most challenging requirements for Operational AI

Operational AI approaches are prone to being overwhelmed by industrial and computational scale. Though one can brute-force model development with custom data science methods, the more significant challenge is scaling models across the hundreds of diverse applications from casting to milling to finishing and the dozens of failure modes for each along with the different root causes of these failures. The computational scale of Operational AI can see thousands of parameters, terabytes of data, hundreds of assets in a single plant. A scalable Operational AI platform must provide a usable, consistent methodology that can be applied against use cases from rotating to static equipment such as winders, rollers, and thermal patterns of caster molding segments.

Perhaps one of the most significant challenges of machine learning, as applied to predictive operations, is estimating the remaining time to a required maintenance or failure event. Detection of a condition that represents an early indication of needed maintenance or a failure is essential, but insufficient to affect manufacturing operations. The operations team needs to know *how long before the predicted event* to adjust operations and maintenance to avoid that event. The time horizon in advance of a critical event is non-trivial to estimate. Furthermore, production operations are dynamic, meaning that the conditions, raw materials, production rates, and steel grades are not static but change with the dynamics of production schedules. To estimate the event horizon for maintenance, a machine learning capability for event horizon estimation is a critical component of an effective Operational AI platform.

## METHODOLOGY

To address the aforementioned challenges, Falkonry's Operational AI methodology includes two sections: a Machine Learning Pipeline and a Service Infrastructure.

### MACHINE LEARNING PIPELINE

In the machine learning pipeline, analysis (learning and inference) and application development occur in the context of a datastream. This concept unifies both historical and real-time data for analysis. A datastream combines signals and entities into an abstract structure representing a use case or a class of equipment. Each entity in a datastream represents a distinct object in that class (*e.g.* compressor) and the approach allows applications to target multiple similar equipment or use cases to be solved with a single application. The analysis of a datastream comprises four distinct stages of machine learning employed in a sequence, as shown in Figure 2, which is the internal structure of an ML application.

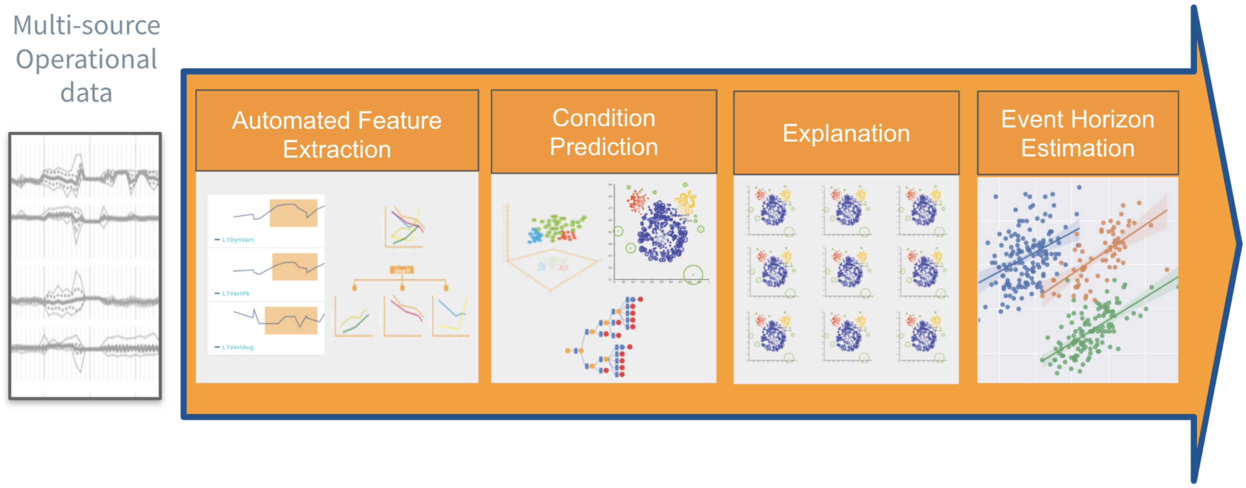


Figure 2: Analysis Pipeline inside an ML application

The output of an Operational AI application is an "assessment". It represents the current condition of an entity in that application and is produced at a desired time interval. An assessment value can be either a condition classification or an event horizon estimate for a given entity and timestamp. The condition classification assessment provides a condition label, the confidence in that label, and the explanation scores for each of the signals that form a part of that datastream. The event horizon estimate assessment provides an estimate value and a confidence interval in that estimate.

Before the Operational AI application can produce near real-time assessments, its machine learning pipeline must first learn a model from training data. The application may be validated by analyzing a distinct data set from that used for training. The four stages of the ML pipeline follow:

**Automated Feature Extraction**

Operational AI is intended to extract information from signals produced by the operation of systems and their effect on the resources consumed in the operation. This operation is observed in signals that are collected by Operational Technology. Falkonry's Operational AI operates on patterns present in such multivariate data. An operational pattern of an entity at time $t$ comprises the trends in multiple signals of that entity as seen over a period of time leading to at a point $t$. We can treat this pattern as a condition.

Falkonry's Operational AI discovers operational patterns and this process depends on automated feature extraction. An unsupervised approach to learning features is used, with the goal of producing a feature vector at each required time $t$ to represent the pattern at time $t$ without knowledge of downstream clustering or classification. The feature vector should consider shapes and waveforms in each of the signals and minimize dimensionality for efficient processing. The feature extraction approach adapts to vagaries of industrial data production such as irregular sampling, multiple independent data sources, noise, gaps and compression as well as different data types - numeric and categorical.

The stage is based on a composition of standard signal processing techniques. This stage uses adaptive windowing to optimize the use of data history. The stage can take advantage of signal metadata provided by the user (such as data type or maximum allowed sampling interval) but does not require algorithm selection by the user. This design approach to automated feature extraction eliminates expensive data preparation efforts that are repeated for each application.

**Condition Prediction**

This stage aims to produce a condition value at every required time $t$ based on the feature vector produced by the automated feature extraction stage for that time $t$. The reported condition at time $t$ may be one of the following: 1) a user supplied label, 2) a system generated label, or 3) unknown. To do so, it uses two additional pieces of information - labeled events and the desired degree of generalization. Each labeled event included for condition prediction, also called a fact, carries a start and end time as well as a condition label. Facts may be either ground truth or hypothesis to be tested. The desired degree of generalization controls the degree of tightness of match between supplied labeled event and patterns arising at other times and results in a choice between high reliability or high predictability. This approach is referred to as semi-supervised learning and produces a condition value even when no example data is provided.

During this stage, feature vectors are first clustered dynamically and a cluster identity is attached to every feature vector. This first step does not depend on the time order of feature vectors and can be called time-free. In the second step, feature vectors supplemented with cluster identity are combined with facts in a classifier to produce a percentage match for each feature vector to the feature vector for each of the fact labels. The result of this stage is further resolved down to a single condition label by using the generalization factor and to isolate anomalies into the unknown condition. The highest match percentage is used as the confidence in the label of the condition value.

The conditions predicted by Operational AI can be selectively reviewed and labeled with the help of experts from the operations team and this stage repeated. In this way, experts' domain knowledge is digitally recorded and is also available for future applications.

**Explanation**

The previous stage outputs most of the condition assessment for the datastream. For every assessment, the datastream also provides a measure of each signal's contribution to making that assessment's condition label (called an "explanation score"). Explanation scores range from 1 (highest contribution) to 0 (no contribution) to -1 (contradicts the assessment but was outweighed by other signals). Explanation scores are calculated per assessment using the condition label and the feature vector from each signal (internal to the condition models). For each signal, the feature vector ("assessment point") is compared against a sample of the feature vectors used during model learning ("sample points"). The algorithm draws a neighborhood boundary around the assessment point to find the nearby sample points. Among these nearby neighborhood sample points, the ratio of points with the same, vs. a different condition, as the assessment is found. This ratio is calculated for all the sample points (not just the neighborhood). These two ratios are used to calculate the explanation score.

**Event Horizon Estimation**

The final stage in the ML pipeline is Event Horizon Estimation[11]. Event Horizon is defined as the length of time until the occurrence of some target event, such as a system failure or a readiness level. As illustrated in Figure 3, this estimate (shown in orange) fluctuates as operating characteristics change and intervention measures are taken. The estimated time to event, *i.e.,* the event horizon (shown in blue), decreases as target events are approached and resets after interventions. Changes in operating characteristics impact how target events unfold. The reliability of any estimation approach requires a dynamic calculation that accounts for these changes. For example, a change in the recipe being applied in production can markedly increase component wear, causing a decrease in the event horizon estimate.

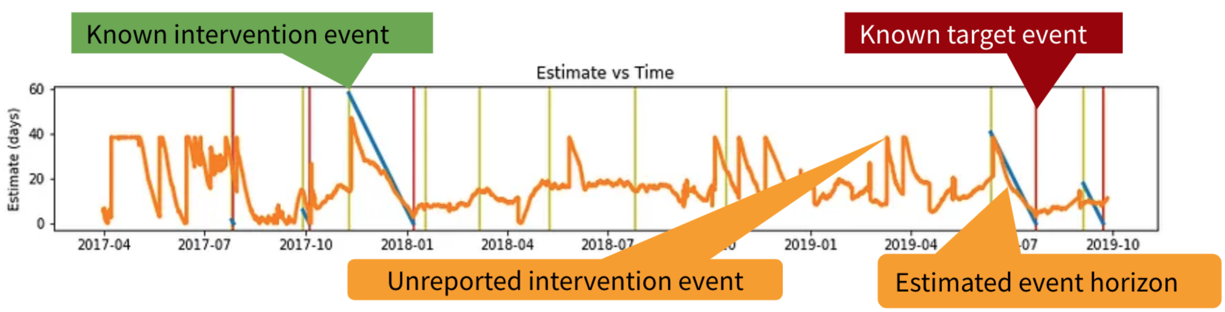


Figure 3: Illustration of event horizon estimation

## SERVICE INFRASTRUCTURE

The Operational AI platform is designed for easy deployment into IaaS (Infrastructure as a Service) environments like AWS or Azure or into privately managed compute and storage infrastructures running a container-compatible Linux operating system. Figure 4 presents the overall architecture of the application deployment.

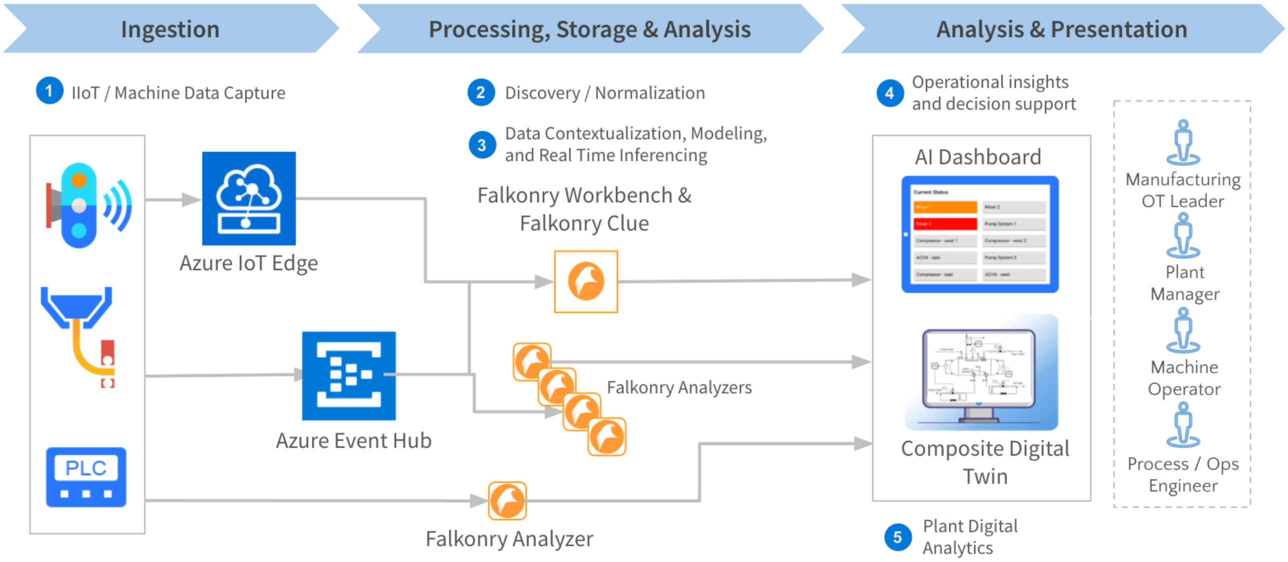


Figure 4: Application deployment architecture

## CONTINUOUS CASTING APPLICATION

### CASE STUDY DESCRIPTION\

As established earlier, existing casting operations are prone to unplanned downtime, affecting equipment reliability and availability. The Operational AI methodology described in the previous section has been applied to multiple continuous casting components, including:

- Mold oscillator - prediction of oscillator failure
- Bending rolls - prediction of bending roll failure
- Dummy bar - prediction/detection of dummy bar misalign and disengagement failure
- Straightener - prediction of stuck roller
- Segment roller - identification of segment roll failure, misalignments, and high roller stress
- Pinch rollers - prediction of pinch roller failure
- Shears - prediction of shear failure

ML applications for over 50 different use cases were developed for this equipment and deployed live in production operations. Table I provides information on selected Operational AI applications for the continuous casting operations. The casting components differ in the nature of their operation. For example, the continuous casting mold oscillator operates continuously, whereas the shears (or scissors) for slab cutting operate periodically. The cars carrying rolled coils operate on demand. The signals data came from a variety of sensors sampling at different rates. Sampling rates ranged from milliseconds to seconds, depending on the operations. With Operational AI, distinct applications for these use cases were deployed live within 18 months, around two weeks for each use case, with one person for data selection and connectivity and two part-time users supporting ML application development and validation.

Table I: Select information on Continuous casting AI applications

| Equipment type | Use Cases | Data Initially Considered | Data Eventually Used | ML Approach | Application Development Time |
|---|---|---|---|---|---|
| Mold oscillator | Oscillator health | 130 parameters 135GB | 1 month, 29 parameters | Unsupervised, Semi-Supervised | 20 iterations over 12 hours |
| Bending rolls | Precursors to jammed rolls | 30 parameters 100GB | 7 months, 9 parameters | Semi-Supervised | 20 iterations over 12 hours |
| Dummy bar | Precursors to failure | 21 parameters 68GB | 6 months, 8 parameters | Semi-Supervised | 20 iterations over 6 hours |
| Straightener | Precursors to jammed rolls | 17 parameters 10GB | 2 months, 15 parameters | Semi-Supervised | 10 iterations over 10 hours |
| Segment roller | Precursors to jammed rolls | 93 parameters 360GB | 5 months, 9 parameters | Semi-Supervised | 35 iterations over 40 hours |
| Segment roller | Segment bulging | | 4 months, 15 parameters | Semi-Supervised | 20 iterations over 30 hours |
| Pinch roller | Precursors to failure | 40 parameters 90GB | 8 months, 8 parameters | Unsupervised, Semi-Supervised | 40 iterations over 50 hours |
| Shears | Motor health | 5 parameters 240 GB | 1 week, 1 parameter | Unsupervised, Semi-Supervised | 5 iterations over 5 hours |

Figure 5 shows the flow of the applied Operational AI-based predictive maintenance solution:

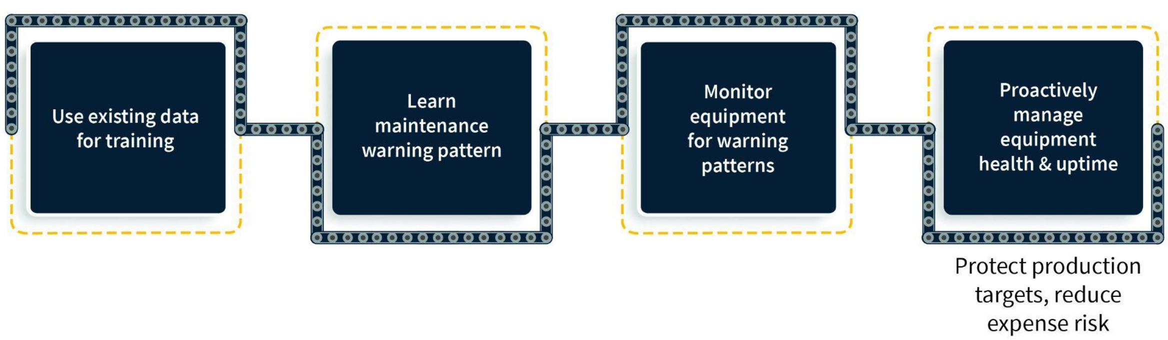


Figure 5: Operational AI-based predictive maintenance solution

**Automated feature extraction:** Signals specific to each stage of the casting process were taken from the plant's data historian. These included current, torque, motor speed, hydraulic pressure, mold level, casting speed, valve status, and others depending on the use case. For each use case:

- Unsupervised learning was automatically performed. This step identified essential patterns of behavior that gave operational experts a better idea of different operating modes.
- A subset of the problem periods preceding events of interest such as part failure, misalignments, or high force events was marked as warning periods to test the hypothesis that these problem periods could be reliably predicted. The Operational AI platform then performs semi-supervised learning, creating models that can discover and distinguish between patterns across supplied parameters to identify conditions that preceded events of interest.

- The ML application, now having learned to detect and recognize the relevant operations patterns occurring before conditions of interest, was validated against data not used in the learning process to validate that it detected the precursor events and therefore could be used to predict the undesirable events of interest.

**Explanation:** The Operational AI platform provides explanations of detected operating conditions by identifying the most responsible parameters for the detected pattern. The ability to understand which parameters are most responsible for the distinct condition arising at a moment in time along with the name of the condition accelerates the troubleshooting and corrective action determination for the predicted failure conditions.

**Event horizon estimation:** After developing a validate early warning application, the next step was to augment condition prediction with time to failure. With one or more historical occurrences of early detection of and the subsequent failure of caster components, the Operational AI platform was used to develop an estimation model for predicting the time to failure (TTF). This estimation of the time horizon before the next failure event, or "event horizon estimation (EHE)," automatically adjusts its estimate as a function of actually observed parameters during real-time operation. EHE enabled operations teams to estimate the time until the occurrence of critical, process-affecting events. Event horizon estimates were used to prioritize interventions, thereby maximizing production time before taking a scheduled maintenance shutdown while also minimizing the probability of incurring an unplanned downtime. While this method is designed to be generally applicable to any use case which has a reliable condition prediction model, work is still underway to verify the reliability of the estimate across different use cases.

## ADVANTAGES

The advantages of this Operational AI approach applied to continuous casting were:

- Simple to use data integration architecture enabled connectivity and streaming of large data quantities from multiple sources with different sampling rates.
- ML applications were created from multivariate data with various, irregular sampling rates, without data engineering effort, for equipment with dynamic operating modes. The automated, consistent, and repeatable development methodology enabled non-data scientist users to create, deploy, and update production-ready ML applications rapidly.
- Operational AI incorporates both unsupervised and semi-supervised ML techniques enabling both discovery of novel conditions and recognition of specific conditions. This method blends learning the patterns of known operating conditions from previously captured operating data while also identifying unexpected and novel conditions in real-time operational data. Therefore, this learning system provides benefits for a wide range of use cases over time and without needing data scientists to develop or maintain these applications.
- ML applications were easily exported for deployment as stand-alone, fully functional, containerized runtime Analyzers deployed on virtual machines or at the edge on gateway processors. The deployment architecture chosen produced real-time predictions predominantly in the cloud, but certain low latency applications were deployed at the edge with constrained resources.

The Operational AI platform handled many different types of sensors, non-preprocessed and irregularly sampled data with multiple time scales, worked with minimal or even no ground truth data and created condition assessments with cause explanations that were found interpretable to the degree that operators could make timely decisions to improve their production systems. These applications run in realistic environments and provide usable real-time insights that address business needs while overcoming the challenges associated with machine learning, ease of adoption, scalability, and deployment.

## RESULTS

The application of Operational AI provided visibility into day-to-day operations, including timely alerts of asset condition to the production operations team. It recovered what would have been production hours lost to unanticipated failures in vertical casting and hot rolling operations. Conditions preceding failure were detected by the Operational AI from one to three weeks in advance of actual failures. Figure 6 illustrates sections of the continuous caster where the application of Operational AI yielded advance notifications to failures.

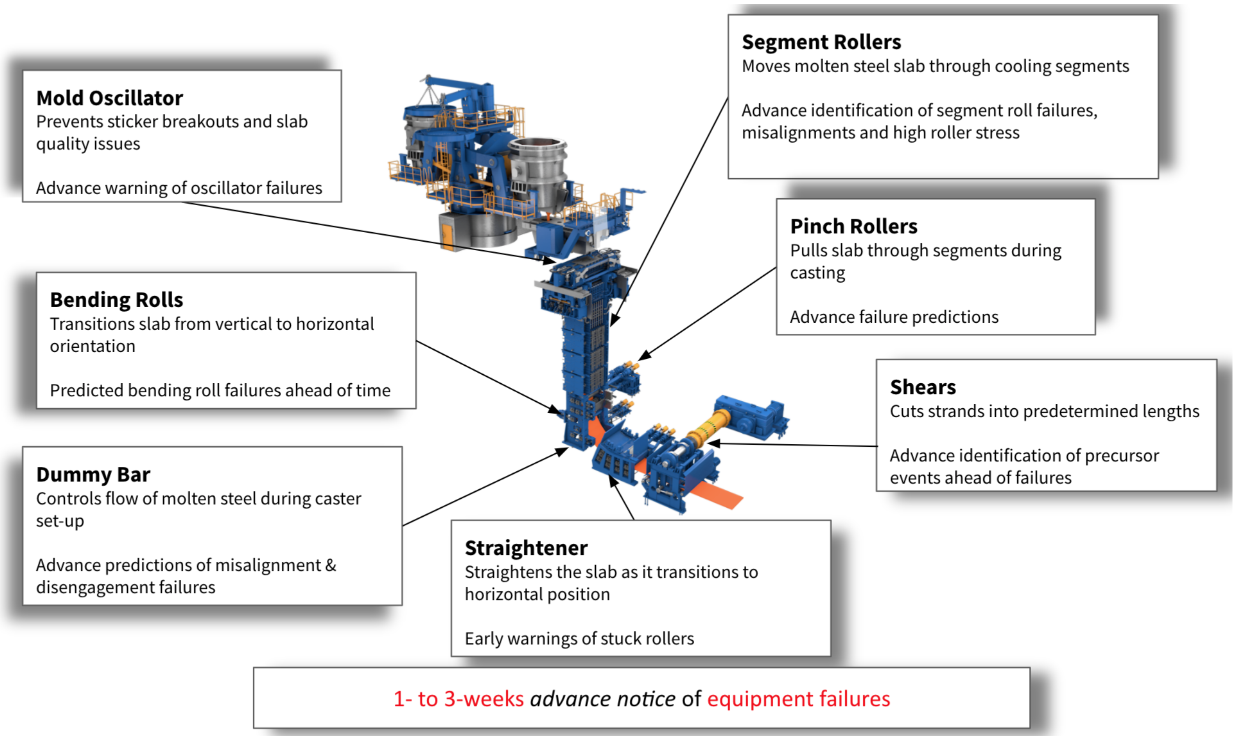


Figure 6: Benefits across ML applications

Figure 7 shows one implementation of Operational AI. The top line in the user interface shows a timeline indicating already scheduled maintenance events. The Operational AI application assessment timeline shows intervals during which detection of conditions known to precede failure were seen. Note that the gap between the occurrence of these conditions is decreasing.

The interface also explains the equipment's condition illustrated by the ranked cause explanation of parameters to the specific application. The legend indicates the relative importance between each parameter during the periods that the early warning conditions were detected. As the failure point approaches in this example, the relative importance of signals increases.

The interface also illustrates the event horizon estimation prediction over time. As the equipment is operating, steel grades and rates of production throughput may change. EHE dynamically adjusts the predicted failure rate over the course of the production operation. The event horizon estimation helps reliability teams make decisions about the priority of early warnings and when to perform maintenance. After maintenance has been completed, the patterns of operating parameters post-maintenance will return to "normal." If, however, maintenance does not resolve the equipment problem or is performed incorrectly, the EHE will not reset, thereby providing a method to confirm correctness of maintenance procedures based on actual production data.

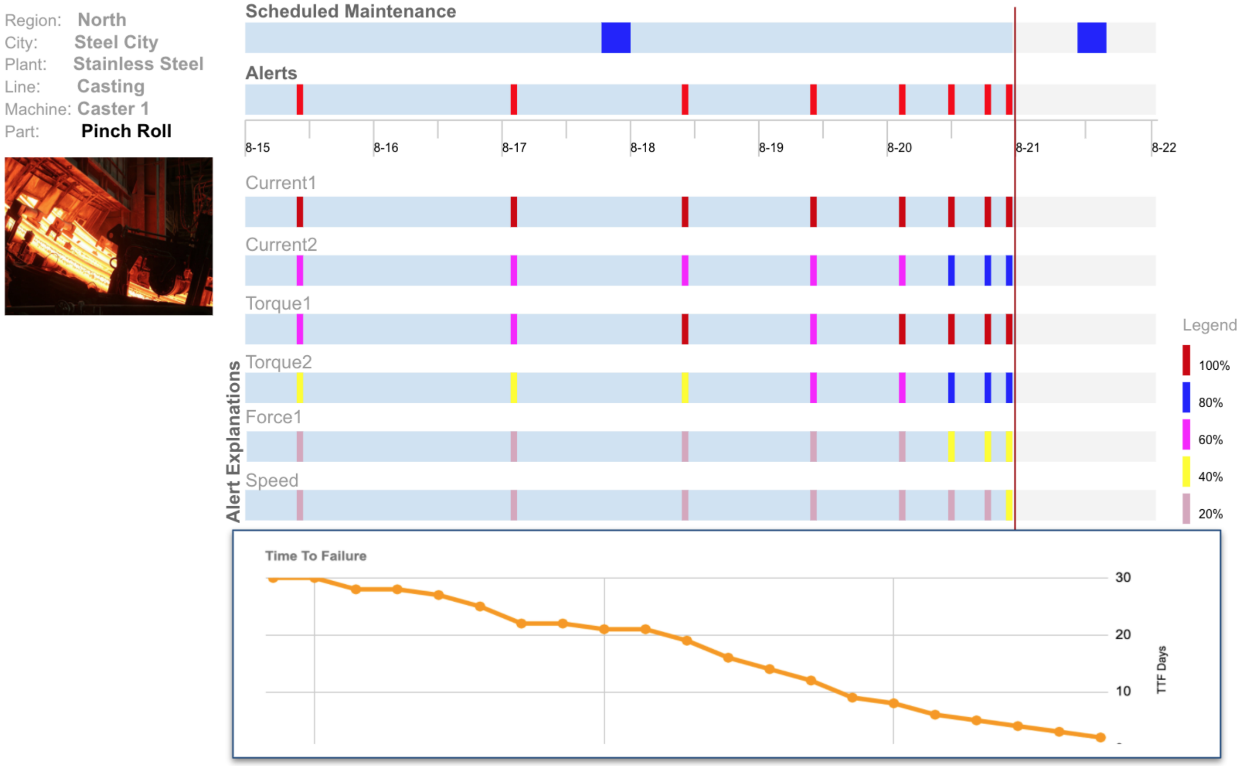


Figure 7: Illustration of early warnings, parameter importance and event horizon estimation

## CONCLUSION

In this work, we discussed the challenges of using AI/ML techniques to improve plant operations. We presented our Operational AI vision and demonstrated a real-world case study in steel manufacturing. We presented the summary of benefits of the steel manufacturing use cases across various equipment with the detailed application-building and deployment approach.

The main conclusions from our work in the space of Operational AI are:

- An intelligence-first approach reduces time-to-first-value. Reducing time-to-first-value enables plant and manufacturing executives who believe in the potential of Operational AI to prove their intuition quickly and with low risk [12].

- The reduction in time-to-first-value removes many hurdles to scaling across multiple equipment, use cases, and time. Plant managers can implement Operational AI without asking for significant corporate resources and plant engineers can discover insights and improve production.

- An intelligence-first alternative to the more conventional data-first approach lowers AI implementation risk, which can often stall value realization. Full deployment of a global data architecture can be deferred until optimization is required to more efficiently capture production value otherwise lost to unscheduled downtime.

- Operational AI makes possible rapid testing and selecting, followed by fast solution deployment, from a wide range of hypotheses. This agile approach of rapidly iterating over models and use cases enables shorter learning cycles and quick identification of essential use cases. Arriving at solutions for use cases faster supports getting stakeholder feedback earlier than conventional methods. This agile Operational AI approach leads to time savings and rapid proof of value.

- Operational AI is capable of learning from available data and improving its predictions over time. It augments the asset owners' abilities by continuously learning and reporting system behavior[13]. It overcomes the time-consuming challenges of gathering new ground truth facts, collecting new operational data sets, learning the new behaviors, and validating the prediction results necessary for traditional ML methods. Manufacturing leaders benefit from digitizing captured expert knowledge so crucial to the application and scaling of Operational AI. Operational AI is cost-effective for discovery, deployment, and scaling; and facilitates ongoing continuous improvement[14].

In this novel way, we are rethinking Operational AI. Operational AI is an intelligence-first approach that reduces business risk and lowers time-to-first value. Operational AI digitally captures expert knowledge for long-term use and enables non-data science experts to quickly develop, deploy, and scale a wide range of ML applications ready to be used by operational practitioners.

## REFERENCES


[1] "Automated machine learning," [Online]. Available: https://en.wikipedia.org/wiki/Automated_machine_learning. [Accessed 2021 April 2021].

[2] C. Lee, "The Intelligence-First Path to Predictive Operations," Falkonry Inc., 6 July 2020. [Online]. Available: https://falkonry.com/blog/the-intelligence-first-path-to-predictive-operations/. [Accessed 22 April 2021].

[3] C. Lee, "Database-First-vs Intelligence-First: The Cart Before the Horse," Falkonry Inc., 10 June 2020. [Online]. Available: https://falkonry.com/blog/database-first-vs-intelligence-first-the-cart-before-the-horse/. [Accessed 22 April 2021].

[4] N. Mehta, "Operational AI: the bridge between Operational Technology and Operational management," Falkonry Inc., 2 December 2020. [Online]. Available: https://falkonry.com/blog/operational-ai-the-bridge-between-operational-technology-and-operational-management/. [Accessed 22 April 2021].

[5] J. Yoon, D. He and B. Van Hecke, "A PHM approach to additive manufacturing equipment health monitoring, fault diagnosis, and quality control," Nantes, 2014.

[6] J. Moyne and J. Iskander, "Big data analytics for smart manufacturing: Case studies in semiconductor manufacturing," *Processes,* p. 39, 2017.

[7] H. Ding, R. X. Gao, A. J. Isaksson, R. G. Landers , T. Parisini and Y. Yuan, "State of AI-based monitoring in smart manufacturing and introduction to focused section," *IEEE/ASME Transactions on Mechatronics,* pp. 2143-2154, 2020.

[8] K. Zope, K. Singh , S. H. Nistala , A. Basak, P. Rathore and V. Runkana , "Anomaly Detection and Diagnosis In Manufacturing Systems: A Comparative Study Of Statistical, Machine Learning And Deep Learning Techniques," in *Annual Conference of the PHM Society*, 2019.

[9] "Operational technology," [Online]. Available: https://en.wikipedia.org/wiki/Operational_technology. [Accessed 22 April 2021].

[10] "iba System," [Online]. Available: https://www.iba-ag.com/en/iba-system. [Accessed 22 April 2021].

[11] N. Mehta, "Event Horizon Estimation - Time to Any Critical Event," Falkonry INc., 17 November 2020. [Online]. Available: https://falkonry.com/blog/event-horizon-estimation-time-to-any-critical-event/. [Accessed 22 April 2021].

[12] N. Mehta, "Rethinking Operational AI," Falkonry Inc., 12 May 2020. [Online]. Available: https://falkonry.com/blog/rethinking-operational-ai. [Accessed 22 April 2021].

[13] C. Lee, "How intelligence first leads to faster learning and cost effective scaling," Falkonry Inc., 13 April 2021. [Online]. Available: https://falkonry.com/blog/how-intelligence-first-leads-to-faster-learning-and-cost-effective-scaling/. [Accessed 22 April 2021].

[14] C. Lee, "Continuous Improvement of an Operational AI Deployment," Falkonry Inc., 18 June 2020. [Online]. Available: https://falkonry.com/blog/continuous-improvement-of-an-operational-ai-deployment/. [Accessed 22 April 2021].