POLITECNICO DI TORINO
Repository ISTITUZIONALE

Quantify production planning efficiency through predictive modeling in manufacturing systems

Original
Quantify production planning efficiency through predictive modeling in manufacturing systems / Monaco, S., Apiletti, D., Francica, A., Cerquitelli, T.. - In: COMPUTERS & INDUSTRIAL ENGINEERING. - ISSN 0360-8352. - ELETTRONICO. - 201:(2025). [10.1016/j.cie.2025.110919]

Availability:
This version is available at: 11583/2997301 since: 2025-02-07T07:37:40Z

Publisher:
Elsevier

Published
DOI:10.1016/j.cie.2025.110919

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

25 November 2026

Computers & Industrial Engineering 201 (2025) 110919



Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie



Quantify production planning efficiency through predictive modeling in manufacturing systems

Simone Monaco [a],*, Daniele Apiletti [a], Andrea Francica [b], Tania Cerquitelli [a]

[a] Department of Control and Computer Engineering, Politecnico di Torino, Torino, Italy
[b] Sandeza, Torino, Italy

ARTICLE INFO

Keywords:
Industry 5.0
Predictive modeling
Production efficiency evaluation

ABSTRACT

This paper proposes a management system designed to evaluate and enhance the optimization degree within manufacturing operations for improved business planning. The proposed model computes predictive data about production forecasts (times, yields, quantity of items produced) to assist operators in filling in these metrics for newly introduced items. It then assesses the discrepancy between the predicted values and the actual measured production data. This assessment aims to provide metrics for evaluating the efficiency of business planning systems, providing a quantified understanding of discrepancies for more accurate profit estimates and strategic planning. The proposed approach exploits shallow and deep machine learning models and transformer-based approaches, and it is experimentally evaluated on a real-world manufacturing dataset. One planned outcome that these metrics will enable is the provision of a tool that supports manufacturing workers by completing data that they cannot define themselves and highlighting potential discrepancies between the manually entered data and the model data, at an early stage of the manufacturing process, thus avoiding errors rather than correcting them afterwards. This approach aims to increase collaboration between humans and machines, in line with the principles of Industry 5.0.

1. Introduction

The fourth industrial revolution has ushered in an era of unprecedented innovation and change in the manufacturing industry. Modern manufacturing companies highly depend on optimizing their production processes to remain competitive and profitable. Production processes are broadly defined and include all steps that are not directly related to machining, such as technical design, industrialization, logistics, and quality processes. This applies, in particular, to companies that do not own the end product and are, therefore, at a lower level of the value chain. Automation and the integration of production machinery have already reduced the potential efficiency gains in pure production time. Greater gains are still possible in indirect processes and planning with a larger human footprint. Machine learning techniques play a central role in this area and go beyond predictive maintenance (Giordano et al., 2022, 2021) to encompass various facets such as quality assessment (Apiletti & Pastor, 2020; Sankhye & Hu, 2020) and optimization of the production processes themselves (Weichert et al., 2019). These techniques serve as transformative tools, not only in predicting potential maintenance needs but also in improving overall production processes. They contribute significantly to improving efficiency, streamlining operations, and raising quality standards in manufacturing.

However, despite the progress that has been made in using machine learning to improve production processes, there is still an urgent need to quantify the extent of this optimization, especially when introducing new products. The challenge lies in developing metrics or frameworks that allow what is known for current production lines to be projected onto new items that may require different production strategies. Consequently, quantifying potential yields and planning investments requires a quantitative measurement of optimization opportunities.

Manufacturers have long been challenged to estimate production figures for new products accurately. This aspect is necessary to provide customers with quotes in advance and plan new productions with previous ones. In traditional environments, these estimates often rely on the expertise and intuition of operators, making them susceptible to human error, subjectivity, and variability in operator skills. However, in the age of Industry 4.0, manufacturing processes generate a wealth of data that can be used to make the prediction of production times more accurate, objective, and data-driven.

One step further, Industry 5.0 represents a paradigm shift emphasizing human-centricity, sustainability, and resilience within smart manufacturing systems (Zhang et al., 2023). Unlike the efficiency-driven focus of Industry 4.0, Industry 5.0 integrates human well-being

* Correspondence to: Corso Castelfidardo, 39, 10129 Torino, TO, Italy.
E-mail address: simone.monaco@polito.it (S. Monaco).

https://doi.org/10.1016/j.cie.2025.110919
Received 20 April 2024; Received in revised form 16 January 2025; Accepted 22 January 2025
Available online 29 January 2025
0360-8352/© 2025 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

as a core element of the production process. Production planning in this context requires addressing the interplay between human flexibility, experience and potential mistakes, machine precision, and advanced information technologies like digital twins, blockchain, and artificial intelligence (AI). These technologies are pivotal in constructing resilient manufacturing systems that prioritize adaptive human–machine collaboration, real-time decision-making, and the integration of human emotional and physical states into the operational workflow. However, challenges such as human-centric task allocation and adaptation of advanced manufacturing models to diverse production scenarios remain significant. Addressing these barriers within the framework of Industry 5.0 requires designing processes leveraging and balancing both human expertise and advanced technological tools.

In line with this perspective, our analysis examines production items through the lens of two different and complementary dimensions: Expert predictions and measured data, which we refer to as *ex-ante* and *ex-post*. The former represents the expected time and resources required for production, while the latter includes simultaneous productions and possible delays due to external practical circumstances. For example, while operators estimate the time required for a particular process, unforeseen problems may arise in real production that extends this period. Conversely, grouping similar productions can rationalize the total time required. While the times of *ex-ante* and *ex-post* are assumed to correlate, this relationship may change due to optimization strategies. By treating these measures separately, an effective optimization approach aims to minimize unexpected delays while maximizing parallelization. In this case, the discrepancy between the times of *ex-ante* and *ex-post* becomes a metric that reflects the efficiency of production strategies. Challenges arise when introducing new items or new processes for existing items. Operators need to forecast actions aligned with existing data, and real statistics only become available after data is collected from these new productions, so no optimization strategy metric is available.

To overcome the challenges of efficient planning, this paper presents an end-to-end approach that uses predictive analytics and actual measured production data to provide a measure of the strength of optimizations in production before the manufacturing process begins. By using *ex-ante* and *ex-post* production data, the proposed management system attempts to predict and subsequently evaluate the effectiveness of the planning optimizations implemented in the production schedules. This paper describes the systematic framework of this management system. It explains its predictive modeling capabilities and subsequent testing of optimization discrepancies to gain actionable insights to improve production planning efficiency.

This research will explore the key methods, challenges and benefits of integrating machine learning into manufacturing processes. We will also discuss the potential benefits of using historical production data to refine our predictive models so that manufacturers can optimize the allocation of resources for production planning and ultimately increase their competitive advantage in the marketplace.

The contributions of the paper are the following:

- We propose an innovative and versatile framework, applicable to different production environments, focusing on manufactured articles, their production steps, and their measures describing the processes, both *ex-ante* and *ex-post*, for which we provide a formal definition.
- Within this framework, we develop two predictive models, both of which process multimodal production data comprising categorical, numerical, and textual information and aim to provide actionable insights for the prediction of production target features.
- Our investigation extends to the application of shallow and deep learning techniques, in particular the use of transformers, to address this prediction challenge. Starting from a real-world use case, we have conducted extensive experiments to investigate the interplay between *ex-ante* and *ex-post* data as predicted by our models.
- We provide experimental results on real-world data from a manufacturing environment, illustrating the relationship dynamics between the *ex-ante* and *ex-post* data representations and providing insights into the effectiveness of our approach and its impact on the production use case.

The paper is structured as follows. Section 2 provides an overview of machine learning applications in manufacturing and highlights the importance of predicting production times for improved planning. Section 3 presents our formalism for describing a production scenario and outlines both the associated *ex-ante* and *ex-post* measures. Section 4 presents the real-world use case, whose experiments are described in detail in Section 5 and demonstrate the potential of predictive models for understanding production processes. Finally, in Section 6, concluding remarks are presented.

2. Related works

Data-driven machine learning techniques are increasingly being used under the Industry 4.0 paradigm, enabling informed decisions in the operational, production, and post-production phases. They can also improve product quality (Apiletti & Pastor, 2020), assembly line efficiency (Hu et al., 2023), customer experience, and inventory management (Kang, Catal, & Tekinerdogan, 2020). This result can be achieved through computer vision-based inspection and monitoring (Junling, Zhang, & Wu, 2020; Koudounas, Giobergia, & Baralis, 2022), defect detection (Andrew et al., 2021; Cerquitelli et al., 2021), and process improvement and optimization (Francesco, Minner, & Battini, 2020).

Regarding optimization, one line of research aims to control production processes by identifying the causes of possible delays, commonly referred to as “disturbances”. Various interpretations and theoretical models have been proposed to analyze their effects in different applications. For example, Xie and Chen (2018) introduced an interval array model for minimizing uncertain workshops to evaluate the delays in the process. In contrast, Rezaei-Malek, Siadat, Dantan, and Tavakkoli-Moghaddam (2019) proposed a product utility function to measure productivity amidst disturbances affecting costs and demand. The latest work utilizes a linear programming model to solve a practical use case, but it fails to fully leverage the generalization capability of machine learning models when applied to new scenarios within the same case study or similar ones. Adane, Bianchi, Archenti, and Nicolescu (2019) developed a system dynamics model to simulate complex manufacturing processes, and Güçdemir and Selim (2017) constructed mathematical models for order fulfillment rate, lead time rate, and delay rate affected by disturbances. Despite these advances, existing models often overlook variations in the robustness of manufacturing systems caused by disturbances. Zhou et al. (2023) suggested a method for addressing the gap in the multilevel modeling and robustness assessment of disturbances in shop floor production processes. However, the proposed models are often very specific and require many parameters to obtain accurate results, making it difficult to apply to new production areas. Recently, Fu et al. (2024) proposed a constraint-driven conceptual design model for new production development. However, it is specifically designed for complex products and may not be systematically applicable in some general settings.

To address these issues, we introduce a novel approach. Instead of modeling production disturbances directly, we use measures from producers’ experience as a proxy for ideal production statistics. A-posteriori deviations from these metrics serve as direct indicators of disturbances and applied optimizations. Within this portrait, various strategies employ different tracked quantities and different tools to track them. Many approaches use simulation tools and industrial Internet of Things (IoT) devices like wearable sensors to collect and analyze real production data. This enables continuous monitoring and analysis of production line performance parameters (Fera et al., 2019). Another method is to use virtual modeling and simulation tools, such as Siemens

NX, to create a virtual model of a production system and simulate and optimize its work (Monica, 2015). While this approach can be effective, it requires an intensive initial effort to develop an effective simulator, and this effort does not scale when moving to meaningfully different production environments. One critical issue with this approach is that off-the-shelf digital twins for each machinery consider the standard machine usage, making it impossible to account for the companies' expertise in improving upon these baselines. By simplifying the portrait that describes production, a practical framework for optimizing production management processes can also be used, which includes a systematic analysis of the trade-off and derivation of a reasonable operating condition (Joppen, von Enzberg, Kühn, & Dumitrescu, 2019). In the context of oil production, optimization involves considering various factors such as reservoir conditions, petrophysics, and PVT data and developing different scenarios to determine the best plan to maximize profitability (Izadmehr, Daryasafar, Bakhshi, Tavakoli, & Ghayyem, 2018).

In Pablo, Samir, Bernard, Robert, and Arnaud (2020), the focus is shifted to Production Planning and Control (PPC), which is described as the essential process responsible for determining the total production quantities (production plan) required to meet commercial targets while ensuring profitability, productivity, and timely delivery. In addition, the PPC includes the management of the production process, enables real-time synchronization of resources, and facilitates product customization (Moeuf, Pellerin, Lamouri, Tamayo-Giraldo, & Barbaray, 2018; Tony Arnold, Chapman, & Clive, 2012) as needed.

Current methods make use of machine learning to quantify the quality of the production process by considering this as a fault assessment task (Wang, Ma, Zhang, Gao, & Wu, 2018). The idea is to consider the disturbances in terms of faults concerning the base production setting, which is assumed to be the optimal one. Then, the problem is shifted from the modeling to which smart sensor infrastructures are required to get a broad picture of the whole process (Zhao et al., 2019). Depending on the data which can be tracked, different machine-learning techniques can fit the task. Some approaches interpreted sensors' general time series output as two-dimensional inputs to be fed into Convolutional Neural Networks (CNN) (Guo, Chen, & Shen, 2016; Janssens et al., 2016). Nevertheless, all these applications are highly customized to fit a very specific case, such as defect diagnosis in bearing (Guo et al., 2016), gearbox (Chen, Li, & Sanchez, 2015), and rotor (Wang, Zhuang, Duan, & Cheng, 2016). Thus, all these approaches cannot be generalized at scale.

From a more general perspective, autoencoders have been studied for unsupervised feature learning. The features learned are then input into a traditional machine learning model for training and classification (Wang et al., 2018). The spiking autoencoder (SNNAE) demonstrates superior performance in monitoring nonlinear processes by reducing fluctuation in fault detection compared to traditional neural networks (Yue, Wang, Zhu, Yuan, & Yang, 2024). Similarly, a vector quantization sparse autoencoder integrates feature extraction and statistical metrics to accurately track system status, showcasing its effectiveness across various complex systems (Gao, Yang, & Jiang, 2022). Collectively, these findings highlight the potential of autoencoders to enhance fault detection and production efficiency in industrial applications. However, all these applications still rely on a significant amount of tracked data from each piece of machinery, making their application on complex and expanding production lines not straightforward. Our method is based on a different premise. We propose to track production processes based solely on the average metrics describing them for various products. We believe this simplification is still valuable for finding patterns and making predictions for new product lines.

Within this context, it is crucial to perform expressive feature engineering as the tracked data can take various forms. Particularly looking at the tracking of production processing addressing different production lines, many textual, numerical, and categorical metadata can have a crucial impact on determining the similarity between different articles.

Common methods for measuring the similarity of textual attributes involve learning representations in an embedding vector space, where the distance between vectors indicates the similarity (Kenter & De Rijke, 2015). For example, Cagliero, La Quatra, and Apiletti (2020) used sentence-based embedding models to link hotel ratings to similarity at a broader level, focusing on cities. Similarly, in the context of manufactured items, several coarse granularities reflect the hierarchical structure of their production steps. In this work, the embeddings are obtained from pre-trained transformer models, which provide valuable results for sufficiently long sentences. However, the names or descriptions of production processes are not always of the same length or detail, making the similarity determination challenging. The following section presents our proposed framework using a real-world use case. Two different approaches to evaluating the similarity of items are described.

3. Problem formalization

Sandeza is a pioneering company in developing management platforms tailored to the industrial sector and aimed at small businesses. Their software solutions monitor the intricate web of manufacturing information required for seamless operations, from production to retail. At the heart of this software suite is zProduction, a fully integrated system orchestrating all aspects of business management and information flow.

The software is based on a central repository that collects information on each item and contains important data on production processes, quality control plans, multimedia documentation and cost analysis. This system streamlines and digitizes the knowledge and experience of everything from managers to production workers to ensure accurate item production. Data collection is done in real-time to ensure data reliability and is facilitated by a user-friendly interface. The integration of manufacturing, logistics, quality and maintenance processes enables real-time monitoring of business performance, including trends, cost analysis and reporting.

Each company item is described within this framework with a detailed set of information. Our primary focus is on the production processes that apply to these company items, which leads us to develop a predictive platform that recommends feature values for newly introduced items based on their similarity to historical affine items. The primary challenge lies in the dual nature of each item's representation: on the one hand, there is real-world data collected by sensors or production workers as soon as production begins, and on the other hand, there are the initial *ex-ante* values estimated by workers at the beginning of each production cycle.

We opted for the zProduction conceptual scheme to describe a general production setting, as there is no clear standard for software producers, particularly in the Italian context with more than 5300 available ERP systems (dod, 2020; Assosoftware.it, 2020). This framework has the advantage of maintaining consistent data structures and processes across all installations in various production environments. In contrast, other alternatives often involve customizations for individual implementations, reducing the generalizability of our downstream approach.

This standardization ensures that the predictive models and approaches we develop are broadly applicable and not hindered by the variability found in customized systems.

We use the term *article* to refer to a manufactured item that is characterized by a certain number of features and labels as defined in the company's specifications. In addition, the production of each article is tied to specific production cycles. An article can be linked to several production cycles, including past cycles. These production cycles comprise a series of phases, essentially the individual steps or processes required to produce the article. The graphical representation of the relationships between items, phases and cycles can be found in Fig. 1.

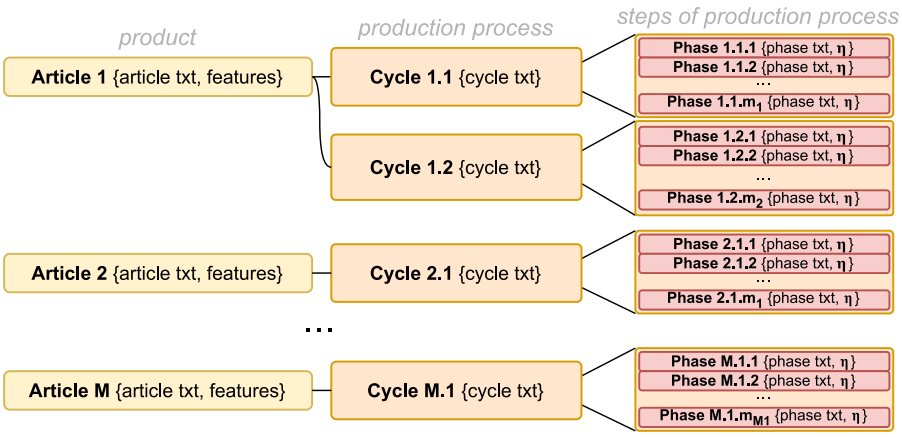


Fig. 1. Pictorial representation of articles, phases, and cycles in the production representation and their associated features. Articles have at least one alternative cycle, each of them made of a list of consecutive phases.

The phases represent the essential steps required for the production of articles and are, therefore, the central point for increasing production efficiency. When introducing a new article, one of the most critical aspects of production planning and costing depends on the metrics that define these phases. These metrics include setup time, teardown time, yield, and items produced per hour.

- **Setup Time** (T_s): The time required to prepare the production equipment and tools before production begins. Setup time is one of the most important elements for production efficiency and scheduling, especially when the production quantity is limited. Therefore, this setup time has a significant impact on the production per unit. Correctly estimating and possibly minimizing these times is crucial for both costs and planning.
- **Teardown Time** (T_t): It is the opposite of setup time, as it focuses on the activities associated with dismantling and cleaning up the production area in order to move to another production facility or finish a production run.
- **Yield** (Y): The ratio between the actual working time and the total time available. A yield of 90% means that in each real-time hour, 54 min are available for productive tasks, while 6 min are used for idle or downtime.
- **Units per Hour** (U_h): Measures the rate at which items are produced within a given time frame, quantifying production speed, and production efficiency. This value represents the actual productivity of units per hour during production, excluding setup and teardown times.

As an example to fix these concepts, we could consider a manufacturing facility producing customized plastic bottles. Different articles represent the types of bottles (e.g., bottles of different sizes or a custom-designed bottle for a specific brand). Each bottle design requires a unique cycle, which consists of several phases necessary for production. One possible cycle could include *Injection Molding*, i.e., forming the bottle shape using molds, *Trimming* and *Cleaning* to remove excess material, *Filling* with the liquid and *Capping* the bottle, *Labeling* by applying a custom-designed label, and finally *Packaging* the final product for shipment. This cycle could be similar to the one of other articles, but the phases would differ depending on the complexity of each particular process.

Focusing on the *Injection Molding* phase, the machine requires setup time for calibration and loading the necessary raw materials, as well as teardown time to remove production waste and clean the machine for the next product. These times can be reduced if, for instance, the previous or subsequent processes use the same raw material. Additionally, the machine may experience idle time during the process. The proportion of time the machine is actively working, relative to the total required time (excluding T_s and T_t), defines the yield Y. Furthermore, factors such as machine performance and external conditions determine the number of units produced during the phase, contributing to the U_h value.

This framework applies to all subsequent phases with appropriate adjustments. By evaluating these metrics for each phase, manufacturers can pinpoint bottlenecks and identify opportunities to optimize processes, thereby improving production efficiency.

New articles require the evaluation of *ex-ante* estimated values for these metrics, which is usually a difficult task. The difficulty lies mainly in correctly taking into account all production factors (people, machines, tools, materials, quantities) and their combinations, especially when the articles produced have significant variability, which is the case for suppliers who do not have production lines. Time pressure adds to this difficulty: this type of company often has little time available to prepare an estimate, both in terms of responding quickly to the customer's request and the cost of the time spent on the estimate. This means that estimators need to define the production cycle, estimate times, define tooling and materials (which may require subsequent estimating from potential suppliers), and define costs and cost breakdowns, often providing these for different batch volumes, in a short time frame.

Once the article has a production history, these metrics are associated with an *ex-post* counterpart generated from the production statistics. While one might expect the statistical data to match the *ex-ante* estimates, our observations show that they often differ significantly in practice.

In the next sections, we will apply our methods to one company's data. Nevertheless, this underlying approach and model are highly generalizable across various manufacturing contexts. The software platform and predictive models are designed as standards, enabling them to be adopted by different manufacturing clients over time. These include sectors managing different kinds of materials (such as plastics, wood, and others), none of which have required different data structures from the previously mentioned ones. However, it is worth noting that the current model may not be directly applicable to chemical or process industries, where continuous manufacturing processes differ significantly from discrete manufacturing (Brierley, Cowton, & Drury, 2006).

The distinction arises from the different purposes and functions of these two types of sets. *Ex-ante* quantities are used to estimate production costs and are therefore associated with an operation that does not consider simultaneous productions and represents a neutral context. In contrast, *ex-post* data comes from a specific scenario in which certain operations — such as setting up equipment for different production cycles — can occur simultaneously, reducing the *ex-post* times for each operation. This real-world measurement captures optimizations and complexities not included in the *ex-ante* data and includes various factors, such as slowdowns due to malfunctions. The

discrepancy between these types of quantities may be due to possible estimation errors in the compilation but also reflects the company's ability to optimize compatible production processes. Although comparing these two variables can be challenging, examining them can reveal inadequacies in either variable and shed light on potential areas for improvement.

The articles, production cycles, and production phases are categorized into different families based on their common characteristics. Each family is accompanied by a descriptive text that provides insight into its defining characteristics. To tackle the task of deriving the appropriate set of numerical attributes for the phases of newly introduced articles, we aim to create a dataset based on these characteristics. This dataset is created using the basic entities detailed in the following sections, where we focus on how the values are distributed in a practical scenario with a representative company. It is worth emphasizing our deliberate decision to retain only the basic characteristics of each entity to allow broader applicability to different business cases.

3.1. Articles

The "Articles" object is a fundamental component within the system, characterized by a textual description indicating the family to which it belongs. Each article is further defined by key–value labels that contain information relevant to the company's business. This information may include production specifications, material composition, dimensions, quality control standards and other data elements relevant to the efficient management of the articles.

The text description, which also contains the family reference, provides a contextual background that facilitates the quick identification and categorization of articles. Families are assumed to be limited and relatively stable over time. An example of the representative company's article families is shown in Fig. 2(a). Meanwhile, the key–value labels provide a structured but elastic representation of the essential details that allow the company to categorize production with a common language efficiently. Fig. 2(b) shows insights into the cardinality of the labels and their assumed values for the representative company.

3.2. Cycles

The "Cycle" component of our system is the bridge between the article phase and the production phase. Each cycle is clearly assigned to a family and thus reflects the categorization framework used for articles. Cycles encompass multiple production phases and represent the detailed step-by-step process of producing an item. In our approach, we recognize that an item may be associated with more than one cycle, each with its own phases.

We consider each combination of "article_cycle" as a separate entity within our data. With this approach, we recognize that even less common cycles can provide valuable information to refine predictive models. This approach ensures that our dataset is sufficiently robust and adaptable to different production scenarios.

3.3. Phases

The "Phases" in our system are the building blocks of the production process and represent the specific steps and tasks required to produce items. Each phase is categorized by a family that appears with its textual description, analogous to the structuring of articles and cycles. The family designation for phases helps to group related tasks and processes and simplifies the management and analysis of production operations.

Phases are denoted by a vector $\bar{\boldsymbol{\eta}} \in \mathbb{R}^{2n}$ of numerical features, which we recall as

$$\bar{\boldsymbol{\eta}} = \eta^{\alpha} \oplus \eta^{\rho}, \tag{1}$$

being η^{α} and η^{ρ} the subvector of the n relevant features separated into the *ex-ante* and *ex-post* ones. In our particular use case, we have η^{*} of the associated four key features: setup time, teardown time, yield, and unit per hour. This formalism represents, for instance, the *ex-ante* and *ex-post* setup time as T_s^{α} and T_s^{ρ}, respectively.

4. Methods

Building on the context presented in the previous section, we would like to develop a framework that automatically estimates production efficiency. Fig. 3 shows its core elements, which we will define in detail. We start with a database that collects relevant metrics associated with the stored production phases. These metrics are organized as *ex-ante* and *ex-post*. We assume that both are valid, since human experts manually validate the former, while the latter are the aggregated result of a statistically significant set of measurements of the same process. This is a strong assumption, but we expect to achieve this goal in a controlled situation after correcting anomalous data in the database and adopting the system for a certain time.

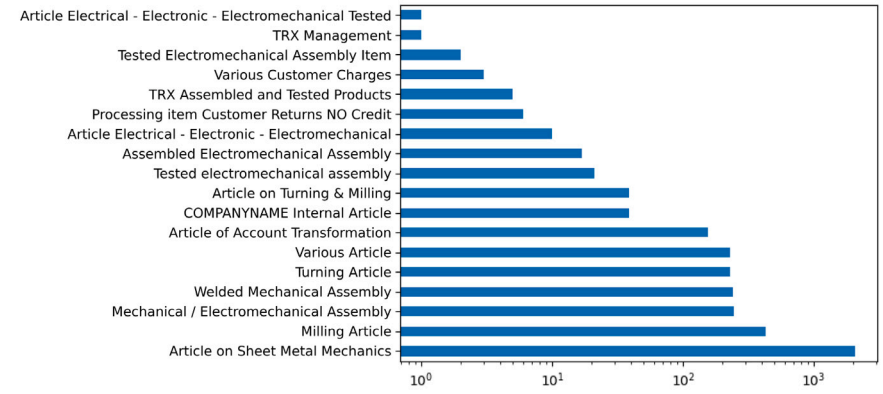
The process of automatic prediction begins when a new article is added to the database. It can be summarized in the following steps.

1. **Data acquisition:** The zProduction system's data collection platform enables manual insertion of the new *ex-ante* metrics. The software is considered a black box within the pipeline and can easily be replaced by any data collection platform.
2. **Predictive model based on *ex-ante* data:** The previous step is supported by automatic suggestions provided by machine learning algorithms trained on the previously available data. The system stores the fields entered by the operator, who can either accept or correct the suggested values. In this way, the procedures for entering new items are accelerated without compromising the quality of the stored values. This model is the first of the two AI-enhanced blocks of the platform, which are presented in detail in the following sections.
3. **Predictive model based on *ex-post* data:** Before each production phase, a second AI-powered block predicts the data associated with the newly added article to provide an expectation of the metric that will characterize it in production.
4. **Evaluation of scheduling optimizations and possible inefficiencies in execution:** The products are actually sent to production, where planning strategies and inefficiencies characterize the values of the *ex-post* phase metrics.
5. **Analysis and improvement:** By exploiting *ex-ante* and *ex-post* values, the system enables analysts and decision-makers to evaluate the effectiveness of their planning strategies. This process makes it possible to identify weaknesses or inefficiencies in the business schedules and make changes to optimize the processes further.

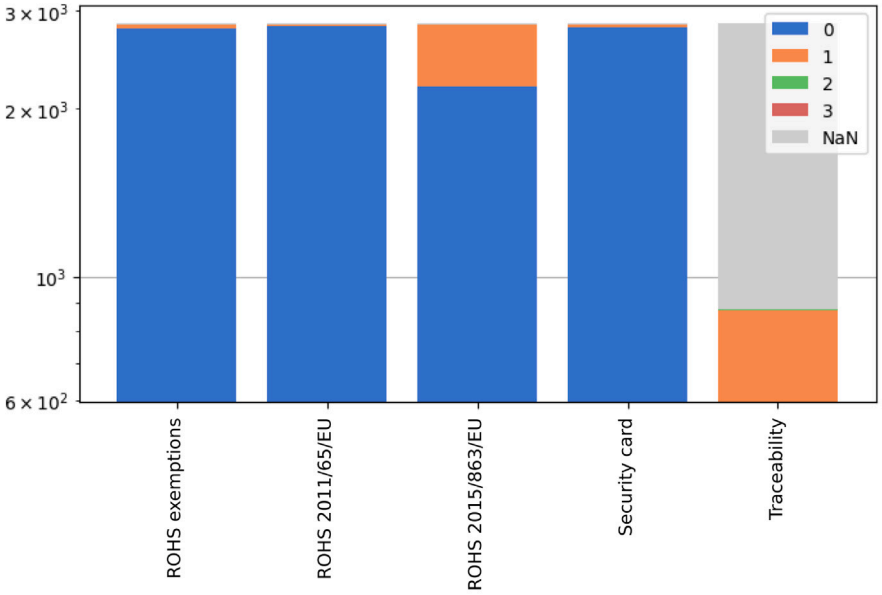
To test the proposed framework, we will look at the practical use of the two AI-powered blocks in the following sections. The first focuses on analyzing a practical use case detailing how these implementations work in real-world scenarios. This approach aims to demonstrate the practicality and effectiveness of integrating AI-powered blocks into operational frameworks, validating their relevance and potential impact on processes. Both models are designed to deliver results that can be validated quickly. This should allow them to be retrained each time a new article is added to the platform.

However, this solution could lead to unnecessary computational overhead, especially in cases where many items are frequently added. To cover these cases and optimize this step, we propose a model update strategy triggered by a real-time estimation of model degradation, as proposed in Cerquitelli, Proto, Ventura, Apiletti, and Baralis (2019).

Building on the process described above, our framework is based on two key theoretical concepts. First, the intuitive idea is that the difference between the estimated and actual product production time directly measures process efficiency. This approach allows us to continuously



(a)



(b)

Fig. 2. Distribution of article families (a) and cardinality of the extra categorical features for articles (b).

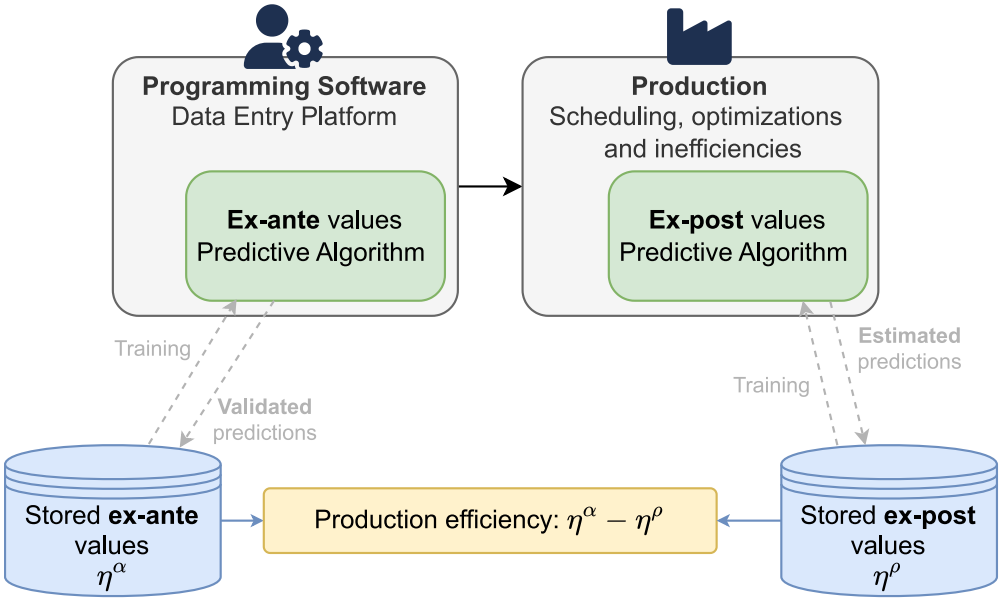


Fig. 3. Production efficiency estimation framework.

Table 1
Dataset statistical description.

	Type	Missing	Unique	Mean	stdev
Phase type name	Text	0.0%	148	–	–
Cycle type name	Text	0.0%	9	–	–
Article type name	Text	0.0%	18	–	–
ROHS exemptions	Cat.	23.18%	3	–	–
ROHS 2011/65/EU	Cat.	23.18%	4	–	–
ROHS 2015/863/EU	Cat.	23.06%	4	–	–
Security sheet	Cat.	23.19%	3	–	–
Traceability	Cat.	77.39%	3	–	–
Ex-ante setup time	Number	40.29%	–	1294.05	1 929.22
Ex-ante teardown time	Number	99.8%	–	1321.20	1 419.00
Ex-ante yield	Number	0.0%	–	90.018	0.42
Ex-ante unit per hour	Number	1.51%	–	238.99	1 452.42
Ex-post setup time	Number	40.41%	–	818.90	2 776.01
Ex-post teardown time	Number	40.41%	–	20.01	153.75
Ex-post yield	Number	40.52%	–	98.19	6.55
Ex-post unit per hour	Number	40.52%	–	2913.91	22 881.09

refine the production strategies by comparing the planned metrics with the real results. Secondly, the strong generalization capability of machine learning models underpins our approach. These models can make valuable predictions for new samples, assuming that these new elements come from the same underlying distribution as the data used for training. This assumption makes perfect sense within a single manufacturing organization where processes and conditions remain consistent (Wang et al., 2018).

The proposed system not only quantifies the efficiency of production planning, but also offers several practical benefits for manufacturing companies. First, it helps to validate or propose accurate estimates of production times, which can significantly improve planning and cost estimation processes. This is particularly beneficial for small and medium-sized enterprises (SMEs) as it provides more accurate cost estimates, thereby improving their competitiveness. Secondly, the system can speed up the preparation of these estimates and drastically reduce the time spent on this task. Conventional approaches often rely on simulation tools that focus only on internal operations and neglect packaging, cleaning and external operations. Our method, however, is independent of specific processes and uses the company's historical data and expertise to provide a more holistic and accurate prediction.

However, there are potential obstacles to implementation. For the system to work effectively, a comprehensive historical data set is required. The more variable the production process, the longer it could take for the system to deliver reliable results, as micro-clustering is required. Finally, it must be ensured that the output reaches a certain level of reliability before it is used for decision-making.

The scalability of the approach is robust, as the model is independent of specific employee characteristics and can handle different levels of complexity. For small manufacturers with lower levels of complexity, the system may offer limited benefits. However, for larger manufacturers or those producing unique items, the model remains applicable due to the similarity of the processes. Optimizing the algorithms for different scales will further improve the model's applicability in different production contexts.

4.1. Dataset

Based on the conception of the production processes described in Section 3, we design a dataset organized at the phase level that summarizes the features extracted from different hierarchical levels. This summarized dataset includes three text-based attributes derived from family descriptions, a subset of categorical features derived from additional item-level labels, and a total of $2n$ numerical features, which amount to 8 in our scenario. In our specific use case, this dataset comprises 25,258 rows and is described in Table 1.

The diversity of data types necessitates an appropriate approach to process them collectively. This process falls under the name of *multimodal data fusion*. Traditional machine learning models handle numerical and categorical features effectively, often employing preprocessing steps such as one-hot encoding for categorical data. However, processing textual features is typically more challenging due to the need for sophisticated preprocessing techniques, which can sometimes result in a significant loss of information (Li, 2018). This limitation arises because traditional models are generally designed to process only a single modality.

In contrast, modern deep learning architectures, such as transformers (Vaswani, 2017), have demonstrated remarkable success in analyzing text and integrating multiple modalities, including text and images or images and audio (Gao, Li, Chen, & Zhang, 2020). For our task, multimodality is simplified, as categorical features and numerical data can, in principle, be represented as text. This approach effectively reduces the problem to a single modality. However, two important subtleties must be considered:

1. While the process appears straightforward, even transformers require careful handling to process numerical features appropriately (Gorishniy, Rubachev, & Babenko, 2022).
2. Deep learning models, despite their ability to handle multimodal data, typically demand large datasets to perform effectively, and their success is not guaranteed when data is limited.

4.1.1. Target features analysis

At the first stage of our dataset inspection, we plot the distribution of target features to show which have the potential to be predicted or which may need a pre-processing step. When available, we analyze the paired *ex-ante* and *ex-post* variables to determine whether the underlying information can be used to adjust for potential anomalies. Some of the following features have very log-tailed distributions, which we considered as outliers and neglected for visualization purposes and to develop more robust algorithms for the remaining part. Features with these characteristics are retained up to the defined upper limit, the 85th percentile of the respective distribution.

Setup Time. T_s features exhibit approximately 40% missing values each, with 13.7% of samples lacking both. As the *ex-ante* annotation remains optional within the platform, the absence of these features is often attributed to compilation errors. Another possible explanation is that certain operations do not require setup time, so the operators leave fields empty instead of assigning null values. Fig. 4(a) underscores this intuition and shows a distribution of *ex-post* values that tends towards zero, a trend that is absent in the *ex-ante* counterpart.

In response, we use *ex-post* values to identify cases where missing *ex-ante* values could instead represent null values. Consequently, we expand the dataset by replacing missing values with zeros, increasing the number of samples by 2000 units (as indicated by the shaded distribution in Fig. 4(a)). Furthermore, the distribution indicates a high degree of discretization in *ex-ante* times, which provides an opportunity to investigate the impact on classification algorithms.

The difference between these measures lies in their purpose: the *ex-ante* value is a typical value set as a worst-case scenario estimate for planning to meet deadlines. Setup, which primarily involves human tasks such as providing resources and machinery, can vary due to production sequencing and is therefore central to planning but difficult to predict prior to production. As it is a value that cannot be measured by sensors or automation but depends strictly on the operator's records, human error can also contribute to the variability of the data, especially with short setup times, making it difficult to tell the difference between setup and production time.

Teardown Time T_t shows an atypical pattern in Fig. 4(b). Similar to the previous scenario, there are a large number of missing values in the *ex-ante* values, which now exceed 99%. Filling these fields with zeros would result in a column that is predominantly filled with zero values, reflecting the *ex-post* truncation time. Applying machine learning techniques to highly skewed distributions could lead to remarkably

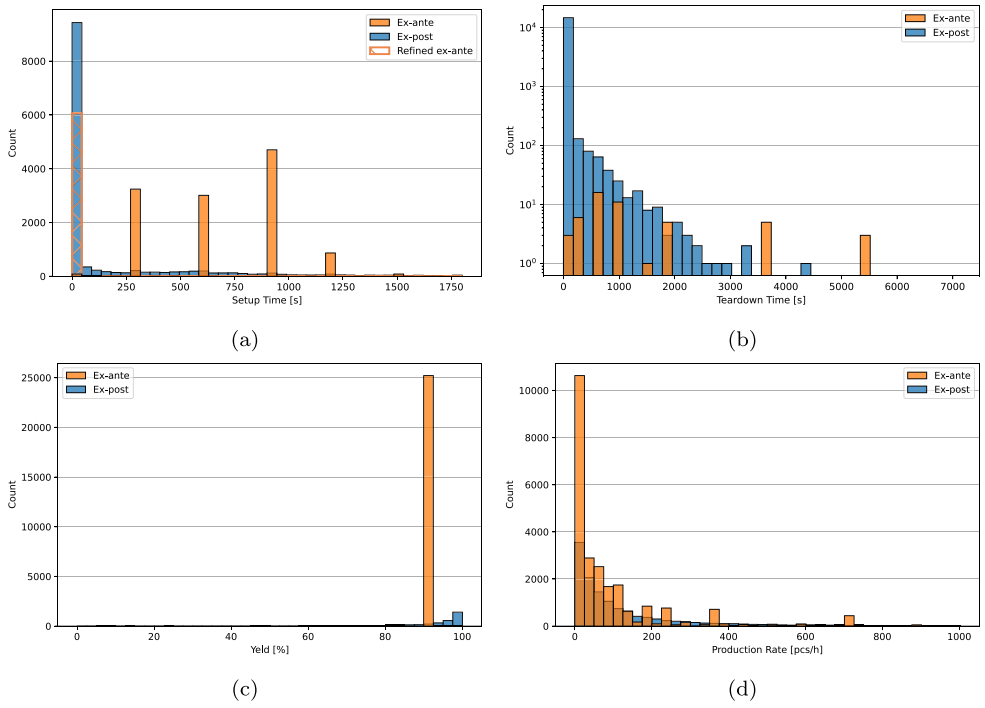


Fig. 4. Comparisons between ex-ante and ex-post distributions for the company numerical features.

poor results. Moreover, we attribute this observation to the likelihood that this feature is only of minor importance for the analyzed company. Therefore, we will disregard these features in the following sections.

Yield. The Yield feature presents a scenario in which its *ex-ante* counterpart has only two possible values, with an overwhelmingly dominant value, accounting for an incredible 99.83% of the dataset. Such a skewed distribution strongly favors one category and indicates an extreme imbalance. Given this imbalance and the observed nominal variability, the *ex-ante* values do not provide meaningful diversity or informative content for the analysis. Given the limited variability and minimal contribution to the information content of the dataset, the yield feature is therefore excluded from subsequent analyses and model considerations.

The *ex-ante* yield value denotes the expected rate of intrinsic disruptions in resources, machines and human factors (e.g. cleanings, breakdowns, etc.). While this value is applied evenly to the theoretical production hours in the ex-ante scenario, in reality interruptions occur at specific points in time, which can vary considerably within individual production batches. This discrepancy between longer-term averages and batch-level events is crucial for assessing the actual statistics.

Unit per hour. The ex-ante U_h provides extensive information with few missing values and a wide distribution. However, a notable peak at zero is a cause for concern as it contradicts the conceptual meaning of the variable representing the number of pieces in a unit of time. We will exclude these cases from further analysis as they may be related to missing values in other areas. In this way, future predictive models can gain advantages when trained exclusively on the remaining part, which is about 75% of the whole dataset. In contrast to the ex-ante distribution, the ex-post distribution does not have a peak at 0 but instead has a slightly flatter curve running towards higher values.

4.1.2. Textual features analysis

Based on the objectives identified in the previous section, our goal now shifts to examining the predictive potential of the remaining features. For the text features, the first strategy is to treat the families associated with phases, cycles, and articles as labels for clustering the target measures. The quality of the clustering result using these labels serves as an indicator, as better clustering results indicate the potential of these families to reveal patterns within the target features.

Table 2
Clustering performance of the textual features over the numerical ones.

	Silhouette score
Phase type name	−0.572
Cycle type name	−0.566
Article type name	−0.567

We calculated the silhouette score for different clusterings based on the associated family. Table 2 reports that the score consistently yielded strongly negative values. This suggests that the inherent variability within each family is considerable, making it difficult to create distinct and self-contained clusters based on family labels alone. Essentially, the family assignments alone do not provide clear groupings for the target variables, emphasizing the complexity of the prediction task in our particular context.

The feasibility of such a naive approach would have opened up the possibility of treating each family as an independent categorical label without dealing with the semantics of its content. Instead, the failure of this approach underscores the need to use a machine learning approach to distil the essential insights while utilizing all available information.

The textual descriptions of the dataset are often limited in length and filled with company-specific acronyms with no general meaning. However, certain families have obvious connections that are likely due to similar words in their descriptions. Applying a naive one-hot encoding to these descriptions would not accurately reflect the different relationships between the labels. Fig. 5 shows the most frequent words, with words lemmatized to their syntactic root for clarity. Noticeably, a relevant proportion of the larger words in the image are predominantly abbreviations or acronyms. To effectively apply deep learning techniques, it is important first to encode these descriptions to cluster similar elements while minimizing the additional complexity. Therefore, we explore three different approaches to tackle this problem, as described in the following sections.

Pure Text Manipulation. In this approach, we dive into pure text manipulation by applying transformer-based models directly to the text components of the dataset. The core of this method lies in leveraging



Fig. 5. Word cloud for Phase type names.

the power of transformers, a class of neural network architectures known for their ability to process sequential data, especially text. Using these models, we aim to capture and understand the complex patterns in the textual descriptions of the families associated with phases, cycles and articles, and then merge them with the other additional information to obtain the final prediction.

We use transformer-based models to uncover hidden patterns, semantic connections and layered structures in the text data. This allows us to better understand the intricate relationships within the dataset. The use of pre-trained language models is promising as they uncover hidden patterns in our text expressions, drawing on an extensive knowledge base from a larger corpus. However, there is a potential drawback, as this approach may entail a significant computational burden without guaranteeing a significant contribution. This can be attributed to the proliferation of jargon and company-specific abbreviations that could affect the model's ability to extract significant insights. In the following sections, we will refer to this dataset as D1.

Feature engineering on similar words. A more classical approach is to extract the relevant features directly from the data. The reason for this is that the textual representations may contain abbreviations or highly technical terminology that language models cannot easily process. This second approach prevents any approach using this type of model and favors a purely data-driven solution based solely on the data itself. After all, families with similar meanings may also contain similar words. This simple assumption can be easily captured by language models or any form of natural language processing embeddings but potentially means an extremely high and unnecessary increase in cost. As an alternative, we propose an embedding for each family name based on multi-hot encoding based on all available and meaningful words (excluding stop words).

A more conventional approach is to extract appropriate features derived directly from the data to overcome the potential limitations of language models when dealing with abbreviations or highly technical texts. This alternative method eschews such models and instead opts for a data-centric solution based solely on the inherent data features. It assumes that families with similar meanings also contain analogous words. Language models or natural language processing embeddings could capture this idea, but their use could lead to an unnecessary increase in computational costs. We therefore propose to generate an embedding for each family name through a multi-hot coding process that uses all relevant and meaningful words (except stop words). The dimension of the embedding space would then correspond to the number of these words, and each description would be a vector where the number one corresponds to all included words and the number zero corresponds to all others.

If we examine the three text descriptions in our data set independently, we obtain embedding spaces of size 28, 11 and 199 for articles, cycles and phases, respectively. Combining these three spaces, we obtain a 238-dimensional space. Our experimental observations using PCA analysis show that condensing the original features to $d = 41$ dimensions is sufficient to capture about 95% of the variability of this space. Consequently, we replace the embedding space with this d-dimensional representation normalized in the range from 0 to 1. This transformed numerical table dataset, denoted as D2, is subjected to analysis using shallow and deep learning algorithms, which are known for their better interpretability and lower computational load compared to deep neural networks.

4.2. Machine learning solutions

To construct the dataset with purely tabular data (D2), we employ a variety of machine learning algorithms suited for structured data analysis. These include:

1. **Random Forests (RF)**: An ensemble learning method that builds multiple decision trees and merges them to improve predictive accuracy and control overfitting. The method operates by averaging multiple deep decision trees T_i, trained on various parts of the same dataset. Namely, given an input x, the Random Forest prediction $\hat{y}$ is obtained as
$$\hat{y} = \frac{1}{N}\sum_{i=1}^{N} T_i(x) \tag{2}$$
This approach minimizes variance and enhances model performance on structured data, making it appropriate for a variety of general-purpose regression and classification tasks. Consequently, we anticipate it will excel in our specific task.
2. **Gradient Boosting (GB):** Gradient Boosting is an additive model that sequentially adds weak learners $h_i(x)$, each improving upon the errors of the previous ones. The prediction $\hat{y}$ at iteration M is given by:
$$\hat{y} = \sum_{i=1}^{M} \gamma_i h_i(x) \tag{3}$$
where γ_i is the learning rate, controlling the contribution of each weak learner. Each learner is trained to minimize the residual errors from the previous learners, thereby "boosting" the model's accuracy.
3. **XGBoost:** It is a specific implementation of gradient boosting with additional regularization to improve performance and reduce overfitting. The objective function for XGBoost combines the loss L over all training samples with a regularization term Ω on the complexity of the trees:
$$\mathcal{L} = \sum_{i=1}^{n} L(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(T_k)$$

where $L(y_i, \hat{y}_i)$ is a reconstruction loss (e.g., the squared error) for regression tasks. The regularization term $\Omega(T_k)$ penalizes the complexity of each tree T_k, promoting simpler and more generalizable models. This model improves upon this approach by utilizing advanced regularization techniques and optimized computations, resulting in enhanced efficiency and accuracy, particularly with large datasets that exhibit complex relationships. Therefore, we anticipate it will perform exceptionally well in our task.

4. **K-Nearest Neighbors (KNN):** A non-parametric algorithm classifying a sample based on the majority class of its nearest neighbors in feature space. KNN relies on calculating the distance between data points, typically using Euclidean or Manhattan distance, making it effective for datasets with distinguishable clusters. KNN classifies an input x by finding the K closest points in the feature space, based on a chosen distance metric (e.g., Euclidean distance). The predicted class $\hat{y}$ for x is the majority class among its K nearest neighbors:

$$\hat{y} = \text{mode}(y_{(1)}, y_{(2)}, \ldots, y_{(K)}) \tag{4}$$

where $y_{(i)}$ represents the class label of the ith nearest neighbor. In regression tasks, the prediction is instead the mean of the K nearest neighbors' values:

$$\hat{y} = \frac{1}{K} \sum_{i=1}^{K} y_{(i)} \tag{5}$$

In this study, KNN effectively identifies local patterns within structured data, serving as a simple but powerful baseline for assessing the performance of more complex models. However, as the size of the dataset increases, the complexity of the model also rises, which can make it less suitable for handling very large datasets.

5. **Multilayer Perceptron (MLP):** An MLP consists of multiple layers of nodes, where each node in a layer applies a non-linear activation function σ to a weighted sum of inputs from the previous layer. For a two-layer MLP with input x, weights W_1 and W_2, biases b_1 and b_2, and activation function f, the prediction $\hat{y}$ is:

$$\begin{aligned} h &= \sigma(W_1 x + b_1) \\ \hat{y} &= W_2 h + b_2 \end{aligned} \tag{6}$$

Here, h represents the output of the hidden layer and $\hat{y}$ is the final output. Multiple layers of this form can be stacked together to form *deep* neural networks, making MLPs able to learn complex, non-linear relationships within the data. We expect this solution to present a powerful expressivity in representing non-linear relationships between inputs and our desired outputs, but it could be harder to train if the data is insufficient.

Instead, for the D1 design of the dataset, we employed TaBERT (Yin, Neubig, Yih, & Riedel, 2020), a model specifically designed for tabular data that includes textual components. TaBERT is a transformer-based architecture that jointly pretrains on natural language text and tabular data, allowing it to learn representations that capture free-form text and structured table schemas. Such a model has been shown to significantly improve tasks requiring joint reasoning over text and tabular data by applying BERT-like masked language modeling to tables, leveraging both row-wise and column-wise attention mechanisms. This approach enables the model to extract meaningful patterns from textual descriptions alongside structured numerical data, which is crucial given the abbreviated and domain-specific terms prevalent in our textual dataset.

Each of these models was selected for its ability to effectively capture the complexities and variances present in the dataset, ensuring strong predictions to facilitate optimal production planning. Additionally, we conducted a grid search to identify the best parameters for this task, which is illustrated in Section 5.

4.3. Evaluation metrics

In our study, we aim to predict detailed target features of newly introduced articles, which is a regression task due to the numerical nature of the features involved. We use the Root Mean Squared Error (rMSE) to assess the quality of our models. rMSE calculates the square root of the average of the squared differences between the predicted and the actual values and thus provides information on the extent of the errors within the predictions. This metric explains the model's accuracy, with lower rMSE values indicating better predictive performance.

In the case of discrete features such as T_s^{α}, another useful representation would view the task as a classification problem, where the models are asked to answer within a limited set of options defined a priori among those seen in the ground truth labels.

Several evaluation metrics are typically used for such a task, including accuracy, precision, recall and F_1-Score. Accuracy measures the overall correctness of the model's predictions. Precision calculates the ratio of correctly predicted positive observations to the total number of predicted positive observations, emphasizing the model's accuracy. Recall, also known as sensitivity, calculates the ratio of correctly predicted positive observations to the total actual positive observations and focuses on the completeness of the model. The F_1-Score, a harmonic mean of precision and recall, provides a balanced assessment of a model's performance on classification tasks.

To ensure the reliability of our model evaluations, all these metrics are calculated using five-fold cross-validation: The dataset is split into five equally sized subsets, with four subsets used for training and the remaining subset used for validation at each iteration. This process rotates through all subsets so that each subset can be used once for validation. By aggregating the results across these iterations, we obtain a more comprehensive and robust assessment of the model's performance and minimize biases and variance that could arise from a single data split. To avoid leakage between training and test sets, all samples of an article are kept in one fold.

5. Results

In this section, we illustrate the effectiveness of the two AI-driven components within the proposed pipeline in the presented real-world use case. Using identical experimental settings, we tackle the two scenarios involving the prediction of *ex-ante* and *ex-post* values.

The dataset used to train the models is the same as described in Section 4.1, with the difference that when training the models to predict the *ex-ante* features, there are no numerical features at this stage of the pipeline. Conversely, when training models to predict *ex-post* values, we include the corresponding *ex-ante* features if they are available in this stage of the pipeline. To configure the D1 dataset, we utilized a pre-trained Bert model from the Multimodal Transformers library (Shi, Mueller, Erickson, Li, & Smola, 2021). Therefore, we evaluated the performance of GB, K-NN, RF, XGBoost, and MLP on the D2 dataset. We tuned the hyperparameters of these models across various ranges, as outlined in Table 3.

All models were configured for a regression task for a single feature at a time. The experiments were performed with an Intel Core i9-10980XE CPU @ 3.00 GHz and an Nvidia RTX A6000 GPU. The code to reproduce all the experiments is available on https://anonymous.4open.science/r/ai4production-planning . Data can be made available upon request, subject to signing an NDA.

5.1. Ex-ante-values prediction

Table 4 summarizes the comparative analysis of the machine learning models on D2 and the transformer-based models on D1. Both were applied to *ex-ante* values.

The results show the superiority of the models applied to the D2 configuration of the dataset, in particular Random Forest, which

Table 3
Hyper-parameter tuning of the models employed for the D2 dataset.

Model	Parameter	Range	Best value
Gradient Boosting	`learning_rate`	0.01–0.1	0.1
	`n_estimators`	100–500	100
K-Nearest Neighbors	`n_neighbors`	3–15	5
	`weights`	Uniform, distance	Uniform
Random Forest	`n_estimators`	100–1000	200
	`max_depth`	5–30	10
XGBoost	`learning_rate`	0.01–0.3	0.1
	`max_depth`	3–10	6
MLP	`hidden_layers`	1–3	2
	`hidden_layers`	16–64	50 (two layers of 50 neurons each)
	`activation`	Relu, tanh	Relu

Table 4
Regression performance (rMSE and r2) for ex-ante features. Values within brackets denote the standard deviation of the result over multiple runs.

	T_s^α		U_h^α	
	r2 (↑)	rMSE (↓)	r2 (↑)	rMSE (↓)
Gradient Boosting	0.7505 (0.0001)	197.8889 (0.0478)	0.3431 (0.0001)	53.0785 (0.0047)
K-Neighbors	0.7152 (0.0070)	211.4029 (2.5957)	0.2295 (0.0232)	57.4770 (0.8621)
Random Forest	**0.7605 (0.0002)**	**193.8807 (0.0962)**	**0.3553 (0.0002)**	**52.5822 (0.0067)**
XGBoost	0.7570 (0.0001)	195.3196 (0.0001)	0.3497 (0.0001)	52.8101 (0.0001)
MLP	0.7416 (0.0037)	201.4021 (1.4465)	0.3335 (0.0041)	53.4650 (0.1658)
Tabert	0.7356 (0.0022)	203.6926 (2.0134)	0.3171 (0.0081)	54.1148 (0.8872)

achieved an rMSE of 193.563 s and 52.556 for T_s^α and U_h^α, respectively. These models consistently outperform the Multimodal Transformer based model applied to D1.

When we consider the performance of the Random Forest model, we calculate the quality of the result by considering that the two predicted features span a range of 0 to 1750 s for T_s^α and 0 to 210 for U_h^α. This implies a normalized error of 11% and 25%, respectively. The production experts consider these errors to be extremely favorable given the complexity of the task. The lower r2 value for U_h^α compared to T_s^α indicates the increased difficulty of the latter task.

5.2. Analysis on discretized features

Given that certain estimated features may have a discrete distribution of values, we can opt for a classification approach as an alternative strategy to the regression task. In particular, the feature T_s^α has a total of 23 different values. Among them, we have filtered out only those whose cardinality is higher than the 0.5% of the dataset so that we obtain five possible target classes (0, 300, 600, 900 and 1200 s). Table 5 shows the results of the different classification models applied to the discretized feature T_s^α.

For each regression model discussed in the previous analysis, we have provided insights into their performance when used in a classification context. However, the K-neighbors algorithm showed inferior results compared to other methods in this context and was therefore excluded from further consideration. We observed an overall deterioration in the mean squared error when we compared these results with the regression setting of each model. Notably, the F_1-Scores were discrete, suggesting that by considering only the selected classes as available options, the system effectively minimizes the user's corrections in a substantial proportion of cases. This interpretation is supported by Fig. 6, in which the confusion matrices represent the relative distribution of correctly classified samples and highlight the typical errors for the best models in both datasets. In particular, the high values along the diagonals of the matrices indicate successful classification for most instances. However, a small but notable proportion of samples are misclassified, often as classes with significantly different labels— - for example, some samples with the lowest value are incorrectly assigned to the highest. This misclassification is since the classification models do not take intrinsic class distances into account, as they treat errors between close and distant classes equally. In regression models, it is possible to derive equivalent classifiers by breaking down the results to the nearest class label. While Random Forest still gets valuable results, the best models appear to be Gradient boosting (when looking at F_1-Score and rMSE), and XGBoost (for Accuracy, Precision, and Recall).

To summarize, the operator's decision-making concerning classification and regression methods depends on considerations specific to the operational context. While classification models offer higher metrics and even outperform the discretized results of regressors, they reduce the need for manual corrections due to their stable, discrete predictions. The downside, however, is their limited prediction range, which restricts results to predefined categories and potentially limits flexibility in decision-making. Conversely, regression models may require more manual intervention, although they can provide a wider range of predictions.

5.3. Ex-post-values prediction

In the second AI-supported block, we show the result of the models presented in the previous section, gathered in Table Table 6. Among the evaluated models, the Gradient Boosting and Random Forest models show the most favorable performance in predicting T_s^ρ, with the Random Forest model achieving the lowest r2 score of 0.4634 and the Gradient Boosting reaching the best rMSE of 181.3321. For the feature Y^ρ, on the other hand, the Gradient Boosting performs best, both in terms of rMSE and r2 score. Similar to previous observations, the Multimodal Transformer model also shows moderate performance on both features. Finally, Gradient Boosting also achieves the best performance for U_h^ρ and obtains the lowest rMSE value of 142.0164. It is worth noting that the errors associated with the ex-post values of U_h are much higher than those associated with the ex-ante values. This discrepancy may be subject to further analysis by the company. All the results of the ex-post metrics confirm that the configuration of the dataset labeled D2 allows the best performance. Among the various models that can be selected for their valuable performance, Random Forest and XGBoost are the ones that score best on all three metrics.

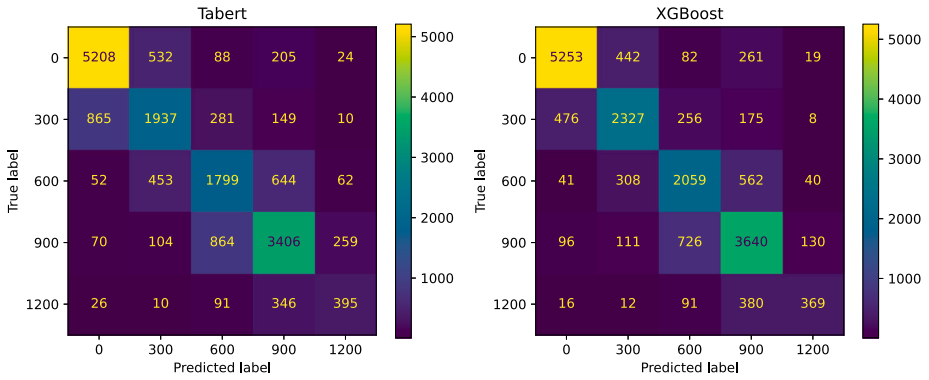


Fig. 6. T_s^{α} confusion matrices of the two best models in the configurations D1 and D2 of the problem.

Table 5
Regression and classification performance for the models trained to predict T_s^{α} in the classification task. Next to the name of each metric, up and down arrows indicate whether a higher or lower value is better. Values within brackets denote the standard deviation of the result over multiple runs.

	Accuracy (↑)	F_1-Score (↑)	Precision (↑)	Recall (↑)	rMSE (↓)
Gradient Boosting	0.7632 (0.0001)	**0.7633 (0.0001)**	0.7647 (0.0001)	0.7632 (0.0001)	**214.937 (0.5104)**
Random Forest	0.7570 (0.0008)	0.7578 (0.0008)	0.7594 (0.0008)	0.7570 (0.0008)	216.803 (1.0342)
XGBoost	**0.7637 (0.0001)**	0.7628 (0.0001)	**0.7653 (0.0001)**	**0.7637 (0.0001)**	217.440 (0.0001)
MLP	0.7497 (0.0037)	0.7501 (0.0039)	0.7530 (0.0039)	0.7497 (0.0037)	220.486 (1.5493)
Tabert	0.6982 (0.0086)	0.7138 (0.0156)	0.7436 (0.0168)	0.7230 (0.0146)	223.040 (2.3401)

Table 6
Regression performance (rMSE and r2) for ex-post features. Values within brackets denote the standard deviation of the result over multiple runs.

	T_s^{ρ}		Y^{ρ}		U_h^{ρ}	
	r2 (↑)	rMSE (↓)	r2 (↑)	rMSE (↓)	r2 (↑)	rMSE (↓)
Gradient Boosting	0.4712 (0.0003)	**181.3321 (0.0460)**	**0.2165 (0.0005)**	**5.7986 (0.0020)**	**0.3928 (0.0003)**	**142.0164 (0.0358)**
K-Neighbors	0.3827 (0.0105)	195.9026 (1.6660)	0.1060 (0.0374)	6.1928 (0.1297)	0.2846 (0.0301)	154.1170 (3.2154)
Random Forest	**0.4634 (0.0008)**	182.6556 (0.1289)	0.1988 (0.0026)	5.8637 (0.0096)	0.3655 (0.0010)	145.1708 (0.1107)
XGBoost	0.4405 (0.0001)	186.5110 (0.0001)	0.1580 (0.0001)	6.0111 (0.0001)	0.3704 (0.0001)	144.6109 (0.0001)
MLP	0.4371 (0.0062)	187.0843 (1.0375)	0.1677 (0.0056)	5.9765 (0.0202)	0.3317 (0.0272)	148.9604 (3.0194)
Tabert	0.4621 (0.0115)	182.535 (1.7453)	0.2097 (0.0093)	5.814 (0.0421)	0.3128	182.537 (3.7531)

5.4. Merits and limitations

The proposed approach is particularly advantageous for small and medium-sized enterprises (SMEs), as it involves lower implementation and operating costs compared to full-fledged simulation models or complete automation systems. Unlike traditional simulation approaches, which require detailed and often expensive modeling of entire production environments, our framework provides a more accessible solution by leveraging existing production data to make accurate predictions. This not only reduces the time and resources required for deployment, but also provides a scalable solution that can be adapted to different production environments without extensive customization.

However, this approach also has its limitations. Its effectiveness depends on the availability of a robust historical database and can cause problems in highly variable production processes where unique or one-off items are often produced. On the other hand, while the system is designed to be generalizable across different production contexts, its application may be less effective in industries with limited production variability, as this structural overhead may be too large. In the future, we plan to develop a precise guide to measure the effectiveness of the proposed approach based on the characteristics of the industrial environment. Despite these challenges, the approach offers a practical and cost-effective alternative for SMEs that want to increase their production efficiency without the overhead of more complex systems.

6. Conclusions

This paper delves into predictive modeling in manufacturing and focuses on the analysis of production data to predict production statistics for items. We present a versatile formalism that can be adapted to different production environments and propose an end-to-end pipeline that provides a measure of the company's optimization strategy based on the difference between the ex-ante and ex-post statistics of each of the company's item steps. At the heart of the system are two AI-powered blocks, each capable of predicting the two sides of the statistics.

In examining these blocks, we look at the effectiveness of machine learning models and transformer-based methods on a real data set. It is noticeable that the shallow learning models, especially Random Forest and Gradient Boosting, show superior prediction performance concerning the target features, which emphasizes their suitability for capturing complex data patterns. Moreover, using a classification approach proves advantages in *ex-ante* measures with a discretizable distribution. Our investigation also extends to testing an MLP to evaluate the impact of deep learning models applied to the dataset in its purely numerical and categorical form, showing that they do not achieve optimal performance. We hypothesize that as the dataset's size increases, applying these techniques could lead to these models performing better than others.

An important observation arises from the discrepancy between *ex-ante* and *ex-post* values, which sheds light on the complexity of the production processes. While regression models help predict precise

numerical estimates, classification methods offer stability and less variability. However, the choice between these methods should be based on the operational requirements and constraints of the production environment.

This study emphasizes the importance of tailored model selection and highlights the trade-offs between precision, interpretability and predictive range. Integrating different modeling approaches and recognizing the nuances between *ex-ante* and *ex-post* data are crucial steps towards improving prediction accuracy in the manufacturing context. Ultimately, this research provides valuable insights for manufacturing companies seeking optimized predictive models tailored to their specific operational needs.

Future works will address (i) the generalization of the proposed approach to other metrics, such as quality control and resource allocation, (ii) domain adaptation to different production environments or manufacturing policies and regulations, and (iii) the integration of human expertise in the modeling phase, for instance, employing reinforcement learning techniques.

CRediT authorship contribution statement

Simone Monaco: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Daniele Apiletti:** Writing – review & editing, Supervision, Project administration, Conceptualization. **Andrea Francica:** Writing – review & editing, Validation, Data curation. **Tania Cerquitelli:** Validation, Project administration, Funding acquisition, Conceptualization.

Declaration of competing interest

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

Acknowledgments

This work is part of the project NODES, funded by the Italian MUR (Ministry of University and Research) under M4C2 1.5 of the PNRR (National Plan for Recovery and Resilience), Italy with the grant agreement no. ECS00000036. The SmartData@PoliTO research centre of Politecnico di Torino, Italy, has partially funded this work.

Data availability

The data that has been used is confidential.

References

ERP - dodigital — dodigital.it. (2020). https://www.dodigital.it/en/erp/. [Accessed 08 August 2024].

Adane, T. F., Bianchi, M. F., Archenti, A., & Nicolescu, M. (2019). Application of system dynamics for analysis of performance of manufacturing systems. *Journal of Manufacturing Systems*, *53*, 212–233.

Andrew, G., Selvaraj, V., Lee, S., Hwang, Y., Lee, K., Lee, N., et al. (2021). Applications of deep learning for fault detection in industrial cold forging. *International Publisher of Production Research*.

Apiletti, D., & Pastor, E. (2020). Correlating espresso quality with coffee-machine parameters by means of association rule mining. *Electronics*, *9*(1), 100.

Assosoftware. it (2020). AssoSoftwareDayPress, sabato 17 ottobre 2020. https://www.assosoftware.it/attachments/article/2919/AssoSoftwareDayPress17102020.pdf. [Accessed 08 August 2024].

Brierley, J. A., Cowton, C. J., & Drury, C. (2006). A comparison of product costing practices in discrete-part and assembly manufacturing and continuous production process manufacturing. *International Journal of Production Economics*, *100*(2), 314–321.

Cagliero, L., La Quatra, M., & Apiletti, D. (2020). From hotel reviews to city similarities: A unified latent-space model. *Electronics*, *9*(1), 197.

Cerquitelli, T., Proto, S., Ventura, F., Apiletti, D., & Baralis, E. (2019). Towards a real-time unsupervised estimation of predictive model degradation. In *BIRTE 2019, Proceedings of real-time business intelligence and analytics* (p. 6). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3350489.3350494.

Cerquitelli, T., Ventura, F., Apiletti, D., Baralis, E., Macii, E., & Poncino, M. (2021). Enhancing manufacturing intelligence through an unsupervised data-driven methodology for cyclic industrial processes. *Expert Systems with Applications*, *182*, Article 115269.

Chen, Z., Li, C., & Sanchez, R.-V. (2015). Gearbox fault identification and classification with convolutional neural networks. *Shock and Vibration*, *2015*(1), Article 390134.

Fera, M., Greco, A., Caterino, M., Gerbino, S., Caputo, F., Macchiaroli, R., et al. (2019). Towards digital twin implementation for assessing production line performance and balancing. *Sensors*, http://dx.doi.org/10.3390/S20010097.

Francesco, Z., Minner, S., & Battini, D. (2020). A supervised machine learning approach for the optimization of the assembly line feeding mode selection. *International Publisher of Production Research*.

Fu, X., Zhang, H., Jing, L., Fan, X., Lu, C., & Jiang, S. (2024). A constraint-driven conceptual design approach for product based on function-behavior-structure design process. *Computers & Industrial Engineering*, Article 109994.

Gao, J., Li, P., Chen, Z., & Zhang, J. (2020). A survey on deep learning for multimodal data fusion. *Neural Computation*, *32*(5), 829–864.

Gao, T., Yang, J., & Jiang, S. (2022). A novel fault detection model based on vector quantization sparse autoencoder for nonlinear complex systems. *IEEE Transactions on Industrial Informatics*, *19*(3), 2693–2704.

Giordano, D., Giobergia, F., Pastor, E., La Macchia, A., Cerquitelli, T., Baralis, E., et al. (2022). Data-driven strategies for predictive maintenance: Lesson learned from an automotive use case. *Computers in Industry*, *134*, Article 103554.

Giordano, D., Pastor, E., Giobergia, F., Cerquitelli, T., Baralis, E., Mellia, M., et al. (2021). Dissecting a data-driven prognostic pipeline: A powertrain use case. *Expert Systems with Applications*, *180*, Article 115109. http://dx.doi.org/10.1016/j.eswa.2021.115109.

Gorishniy, Y., Rubachev, I., & Babenko, A. (2022). On embeddings for numerical features in tabular deep learning. *Advances in Neural Information Processing Systems*, *35*, 24991–25004.

Güçdemir, H., & Selim, H. (2017). Customer centric production planning and control in job shops: A simulation optimization approach. *Journal of Manufacturing Systems*, *43*, 100–116.

Guo, X., Chen, L., & Shen, C. (2016). Hierarchical adaptive deep convolution neural network and its application to bearing fault diagnosis. *Measurement*, *93*, 490–502.

Hu, J., Wang, H., Tang, H.-K., Kanazawa, T., Gupta, C., & Farahat, A. (2023). Knowledge-enhanced reinforcement learning for multi-machine integrated production and maintenance scheduling. *Computers & Industrial Engineering*, *185*, Article 109631.

Izadmehr, M., Daryasafar, A., Bakhshi, P., Tavakoli, R., & Ghayyem, M. A. (2018). Determining influence of different factors on production optimization by developing production scenarios. *Journal of Petroleum Exploration and Production Technology*, *8*, 505–520.

Janssens, O., Slavkovikj, V., Vervisch, B., Stockman, K., Loccufier, M., Verstockt, S., et al. (2016). Convolutional neural network based fault detection for rotating machinery. *Journal of Sound and Vibration*, *377*, 331–345.

Joppen, R., von Enzberg, S., Kühn, I. A., & Dumitrescu, I. R. (2019). A practical framework for the optimization of production management processes. *Procedia Manufacturing*, *33*, 406–413.

Junling, C., Zhang, Z., & Wu, F. (2020). A data-driven method for enhancing the image-based automatic inspection of IC wire bonding defects. *International Publisher of Production Research*.

Kang, Z., Catal, C., & Tekinerdogan, B. (2020). Machine learning applications in production lines: A systematic literature review. *Computers & Industrial Engineering*, *149*, Article 106773.

Kenter, T., & De Rijke, M. (2015). Short text similarity with word embeddings. In *Proceedings of the 24th ACM international on conference on information and knowledge management* (pp. 1411–1420).

Koudounas, A., Giobergia, F., & Baralis, E. (2022). Time-of-flight cameras in space: Pose estimation with deep learning methodologies. In *2022 IEEE 16th international conference on application of information and communication technologies* (pp. 1–6). IEEE.

Li, H. (2018). Deep learning for natural language processing: advantages and challenges. *National Science Review*, *5*(1), 24–26.

Moeuf, A., Pellerin, R., Lamouri, S., Tamayo-Giraldo, S., & Barbaray, R. (2018). The industrial management of SMEs in the era of industry 4.0. *International Publisher of Production Research*.

Monica, Z. (2015). Optimization of the production process using virtual model of a workspace. *vol. 95*, In *IOP conference series: materials science and engineering*. IOP Publishing, Article 012102.

Pablo, U. C. J., Samir, L., Bernard, G., Robert, P., & Arnaud, F. (2020). *Machine learning applied in production planning and control: a state-of-the-art in the era of industry 4.0*. publisher of Intelligent Manufacturing.

Rezaei-Malek, M., Siadat, A., Dantan, J.-Y., & Tavakkoli-Moghaddam, R. (2019). A trade-off between productivity and cost for the integrated part quality inspection and preventive maintenance planning under uncertainty. *International Journal of Production Research*, *57*(19), 5951–5973.

Sankhye, S., & Hu, G. (2020). Machine learning methods for quality prediction in production. *Logistics*, *4*(4), 35.

Shi, X., Mueller, J., Erickson, N., Li, M., & Smola, A. J. (2021). Benchmarking multimodal automl for tabular data with text fields. arXiv preprint arXiv:2111.02705.

Tony Arnold, J. R., Chapman, S. N., & Clive, L. M. (2012). *vol. 118*, *Introduction to materials management*. Pearson.

Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*.

Wang, J., Ma, Y., Zhang, L., Gao, R. X., & Wu, D. (2018). Deep learning for smart manufacturing: Methods and applications. *Journal of Manufacturing Systems*, *48*, 144–156.

Wang, J., Zhuang, J., Duan, L., & Cheng, W. (2016). A multi-scale convolution neural network for featureless fault diagnosis. In *2016 international symposium on flexible automation (isfa)* (pp. 65–70). IEEE.

Weichert, D., Link, P., Stoll, A., Rüping, S., Ihlenfeldt, S., & Wrobel, S. (2019). A review of machine learning for the optimization of production processes. *International Journal of Advanced Manufacturing Technology*, *104*(5–8), 1889–1902.

Xie, N., & Chen, N. (2018). Flexible job shop scheduling problem with interval grey processing time. *Applied Soft Computing*, *70*, 513–524.

Yin, P., Neubig, G., Yih, W.-t., & Riedel, S. (2020). Tabert: Pretraining for joint understanding of textual and tabular data. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 8413–8426).

Yue, B., Wang, K., Zhu, H., Yuan, X., & Yang, C. (2024). Spiking autoencoder for nonlinear industrial process fault detection. *Information Sciences*, *665*, Article 120389.

Zhang, C., Wang, Z., Zhou, G., Chang, F., Ma, D., Jing, Y., et al. (2023). Towards new-generation human-centric smart manufacturing in industry 5.0: A systematic review. *Advanced Engineering Informatics*, *57*, Article 102121.

Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., & Gao, R. X. (2019). Deep learning and its applications to machine health monitoring. *Mechanical Systems and Signal Processing*, *115*, 213–237.

Zhou, Y., Zhou, G., Zhang, C., Chang, F., Wang, Z., & Men, S. (2023). Multi-level modeling and robustness evaluation of disturbances in intelligent workshop with temporal snapshot network. *Journal of Manufacturing Systems*, *71*, 20–33.