Two-Step Textual Similarity-Based Approach for Predicting Suitable Production Line for a Newly Designed Product

Jantana Panyavaraporn[1,*], Chantra Nakvachiratrakul[2] and Paramate Horkaew[3,*]

[1]Department of Electrical Engineering, Faculty of Engineering, Burapha University, Chonburi, Thailand
[2]Department of Industrial Engineering, Faculty of Engineering, Burapha University, Chonburi, Thailand
[3]School of Computer Engineering, Suranaree University of Technology, Nakhon Ratchasima, Thailand.

∗Corresponding Email: jantanap@eng.buu.ac.th, phorkaew@sut.ac.th

Received December 20, 2024, Revised June 4, 2025, Accepted July 7, 2025, Published December 30, 2025

Abstract. *During industry 4.0, digital technology has been integrated with manufacturing processes to improve operational efficiency and hence to enhance organization competitiveness. To this end, computerized methods have been rapidly developed to tackle various production and delivery issues. Production planning and scheduling often demand substantial resources, especially in terms of manpower and time. Consequently, minimizing the time dedicated to both planning and scheduling can hasten product delivery. This paper proposes a novel algorithm that analyzes an unseen process and then predicts a production line by using two-step similarity measures, used in ensemble within the process. Provided with an unseen product model, it identifies the most suitable line, based on the availability of machinery required by the process. In the experiments, eight similarity measures were assessed, based on a realistic production plant. The results revealed that Jaccard similarity and Dice similarity coefficients gave the most accurate predictions. The proposed method is thus believed to be applicable in dynamic production scenarios. Moreover, the developed system also supports incremental production lines.*

Keywords: similarity, predictive analytics, industry 4.0

1. Introduction

The process of designing production lines in response to specialized customer's designs is known as Engineer-To-Order (ETO) manufacturing. The process entails producing the original items using the available resources. To this end, the engineering team has to adjust an existing process to fulfill the unique requirements of such items. Therefore, the challenging complexity of ETO systems calls for deep understanding of production capability and involving constraints. These insights would enhance the production design and the resulting process adjustments [1].

When designing a production process, one must consider various constraints to ensure manufacturing efficiency [2]. For instance, effective production line arrangement can increase production efficiency and reduce waiting times. Both factors are crucial for maintaining competitiveness in today's markets [3]. Additional techniques such as lean manufacturing can streamline the selection and adjustment of the production lines that minimize their downtime [3].

Extensive research has addressed the issue of time spent on production planning. A notable study [4], for instance, proposed reducing production planning time via lean techniques. They focused on production line balancing, which enhanced customer satisfaction and operational efficiency. Another study [5] tackled planning issues by using a nonlinear clearing function model that expressed the relationship between production cycles and time. Similarly, in [6], a novel method, which integrated computer software into production planning and scheduling was described. It focused on improving overall efficiency, while reducing work time.

The concept of industry 4.0 emphasizes the use of modern technologies such as Artificial Intelligence (AI), the Internet of Things (IoT), big data analytics, autonomous robots and cloud computing. Industry 4.0 provides numerous opportunities for enhancing efficiency. They include automation and streamline processes that reduce human errors while improving manufacturing precision. It also integrates technologies such as IoT and data analytics in predictive maintenance. Such integrations enable early issue detection and minimize production downtime, thereby lowering total operational costs. Moreover, the flexibility of Industry 4.0 enables rapid adaptations of existing production lines in response to real-time market demands, such as customizable products and their diverse options. Accordingly, the adoptions of these modern technologies empower an organization to develop new products and services that better align with customer needs, hence gaining leverages to enter a new market more competitively.

Numerous studies have focused on enhancing production pipelines. In fact, the latest advancements in AI and machine learning (ML) have made significant progress in computerized production planning and scheduling. The notable transformations included increased efficiency and being adaptive to dynamic production environments. A number of recent studies emphasize the benefits of reinforcement learning (RL) on the decision-making of production scheduling. Particularly, the ability of RL to address complex scheduling challenges was highlighted in a systematic review on this topic [7]. AI techniques were additionally found integrated with simulation models. The integration was demonstrated in the case studies using Microsoft™ (MS) Project Bonsai and AnyLogic™ and already shown a promising outcome from optimized task scheduling in industrial settings [8]. The increasing needs for adaptable solutions in variable batch production called for multilevel scheduling models. In response to the requirements, innovations such as attention mechanisms and image recognition [9] could improve the scheduling accuracy by more than 30%. Further analysis of this particular study has revealed the roles of scheduling rules and algorithms being essentially shifted toward intelligent manufacturing. Therein, emphasis was placed on the robustness that enabled the algorithms to address practical scheduling issues more effectively. Several studies on computer-based planning systems underscore the value of data-driven decision making for enhancing production efficiency. Such systems driven by AI have proved their effectiveness in real-world

applications, based on standard key performance indicators (KPI) [10]. In addition to AI, ML-based and dynamically controlled project management algorithm was also introduced [11]. The algorithm was developed to optimize production scheduling in environments with unpredictable operation times and complexities. In particular, by leveraging AI and an attention mechanism, the algorithm efficiently collected, extracted, and predicted operation times for products not yet produced. Its potential applications included mixed production lines and those with variable batch sizes. Another study applied multi-objective linear programming to production planning optimization [12]. The method achieved about 16% cost reduction for dynamic planning. Meanwhile, production scheduling under the Industrial Internet of Things (IIoT) platform was also explored [13]. The study addressed the challenges presented in intelligent scheduling for hybrid flow shop by using deep RL. They reported an improvement of over 6% in scheduling efficiency, compared to traditional approaches. Another ML algorithm called M5P was applied to production planning in smart factories [14]. It highlighted the vital roles of IoT data in optimizing the production process and in related decision supports. Last but not least, flexible IoT architecture and AI models were developed for monitoring labor-intensive manufacturing sites [15]. That method facilitated accurate production forecasting and real-time activity recognition. It thereby demonstrated that effective planning could be realized through machine utilization analysis and sequential AI learning from data acquired by IoT.

As evident in [4]-[15], various approaches have been proposed to improve production planning and scheduling. Early works emphasized lean techniques, mathematical models, and software-based tools to reduce planning time and to improve efficiency. With the emergence of Industry 4.0, technologies such as AI, IoT, and big data have enabled automation, predictive maintenance, and adaptive production systems. Recent studies have shown that AI can effectively handle complex and dynamic scheduling problems. Integration with simulation, attention mechanisms, and IoT-based architectures has demonstrated measurable improvements in scheduling accuracy, cost reduction, and real-time decision-making, highlighting the growing role of intelligent, data-driven methods in modern manufacturing. While AI can effectively address complex and dynamic production scheduling problems, this paper instead presents a novel management strategy based on data science principles. The proposed algorithm does not rely on ML and is therefore not affected by learning-related biases. Instead, it utilizes deterministic similarity analysis to evaluate the relevant data. A key advantage of our approach is that, although errors exist in the raw data, they only impact on the input directly and do not propagate or accumulate throughout the learning process.

In this paper, a factory that offers a diverse range of products tailored to customer requirements was studied. The factory operates a mass production system with several sub-production lines, each featuring unique stages and machinery designed to meet specific customer demands. Upon receiving the product design from a customer, engineers develop a customized process, comprising of necessary machinery for a given production line. This information is then passed on to the planning department, whose members then determine which production line has the most compatible machinery to that specific production. Their primary objective is to minimize the time needed for line adjustments due to different compositions. Therefore, this step typically involves comparing and choosing a line that closely matches a design. Unfortunately, with a large number of production lines available, the step can be tedious and time-consuming, especially for less experienced planners.

Following the problem statement, this paper presents a novel algorithm for selecting production lines, given new product models. Each model may require different machinery according to specifically customized designs. Traditionally, there is no sole reliable way to guide such selections or to confirm whether they are optimal. Consequently, adjustments to the machinery within selected lines were often inevitable in real-world scenarios. It is thus anticipated that the proposed method could help reduce the unnecessary workload imposed on the production planning staffs, thereby enhancing the overall efficiency of the production planning and the resulting processes.

2. Similarity Metrices

Similarity or measures or metrices were normally utilized to quantify the degree of similarity between two data points, vectors, or sets. Each metric has unique characteristics and hence is suitable for different types of data and purposes. Listed below are the explanations and corresponding mathematical formulae of similarity metrices employed in this study.

A. Cosine Similarity

Cosine similarity measures the cosine of the angle between two non-zero vectors in an inner product space. It is defined as the dot product of the vector pair divided by the product of their magnitudes. This measure is particularly useful for assessing the similarity between documents or text data [16], as it focuses on the orientation of the vectors rather than their magnitude.

B. Jaccard Similarity

Jaccard similarity measures the similarity between two sets by comparing the size of their intersection to the size of their union. This measure is effective for comparing binary attributes or sets, such as measuring the similarity between two groups of items [17, 18].

C. Euclidean Distance

Euclidean distance represents the straight-line distance between two points in Euclidean space, calculated from their coordinates in multiple dimensions. It is typically used in geometry and spatial analysis [19], particularly when the physical distance between points is of interest.

D. Manhattan Distance

Manhattan distance calculates the distance between two points by summing the absolute differences of their coordinates along grid-like paths. This measure is suitable when movement occurs along axes or grids, often used in urban analyses and path-finding algorithms [20].

E. Pearson Correlation Coefficient

Pearson correlation coefficient quantifies the linear relationship between two variables. The values range from -1 to 1, indicating perfect negative and positive correlations, respectively. The closer the coefficient is to 0, the less likely that these variables are correlated. This measure is normally found in statistical analysis [21] to assess the strength and direction of the relationship between two continuous variables.

F. Dice Similarity Coefficient

Dice similarity coefficient measures the similarity between two sets by considering the size of their intersection relative to the total size of both sets. It is particularly useful in image analysis [22] and document comparison [23], where the emphasis is on the overlap between sets.

G. Hamming Distance Measures

Hamming distance measures the number of positions at which two strings of equal length differ, making it suitable for categorical data represented in binary form. This measure is widely used in coding theory and error detection [24], where the difference between two binary strings needs to be evaluated.

H. Jensen-Shannon Divergence

Jensen-Shannon divergence quantifies the similarity between two probability distributions by measuring the average of the Kullback-Leibler divergence of each distribution with respect to the average of both distributions. It is often employed in statistics and machine learning to compare probability distributions, particularly in tasks involving generative models [25].

The formulae of the above similarity metrices are listed in Table 1, where A and B are the two data sets, whose similarity is measured. Therefore, given a product model described by$\{[N_i]x_i\}$, the proposed method aimed at finding a production line, whose$\{[M_j]y_j\}$ sequence was the most similar, with respect to some metric, where N (or M) and x (or y) are, respectively, the number and the name of an i^{th} (or j^{th}) machine required in the process.

Table 1 Formula of the similarity metrices employed in this study

Similarity (Abbrev.)	Formula	Eq.
Cosine similarity (CS)	$\cos ine = \frac{A \cdot B}{\|A\|\|B\|}$	(1)
Jaccard similarity (JS)	$Jaccard = \frac{\|A \cap B\|}{\|A \cup B\|}$	(2)
Dice similarity coefficient (DS)	$dice = \frac{2\|A \cap B\|}{\|A\| + \|B\|}$	(3)
Euclidean distance (ED)	$d = \sqrt{\sum_{i=1}^{n} (A_i - B_i)^2}$	(4)
Manhattan distance (MD)	$d = \sum_{i=1}^{n} \|A_i - B_i\|$	(5)
Pearson correlation coefficient (PC)	$r = \frac{\sum (A_i - \overline{A})(B_i - \overline{B})}{\sqrt{\sum (A_i - \overline{A})^2 \sum (B_i - \overline{B})^2}}$	(6)
Hamming distance (HD)	$d = \sum_{i=1}^{n} xor(A_i, B_i)$	(7)
Jensen-Shannon divergence (JSD)	$JSD(A\|B) = \frac{1}{2}(KL(A\|M) + KL(B\|M))$; $M = \frac{A+B}{2}$	(8)

3. Data Preparation and Characteristics

This paper addresses the problem of finding the most suitable production lines for a given new product design, so that the adjustments made to the selected lines would be minimal.

The data considered in the process were collected from a set of 84 product design models. Each model was represented by an ordered sequence $\{[N_i]x_i\}$. Example of two models analyzed in this study are illustrated in Table 2. In this example, Model #1 can be interpreted as a design requiring two BAS-326H-0 machines, three S-7300A machines, two GC20618-2 machines, one invert machine, one metal detector, two table inspection machines, one table set front, one computer monitor, and two barcode printers. Likewise, Model #2 was that required four BAS-326H-0 machines, three GC20618-2 machines, one B-8452B-7 machines, one invert machine, one metal detector, one table inspection machine, one auto stamping (long), one computer monitor, and two barcode printers.

Subsequently, data from the 10 existing production lines were also collected. Each line consisted of a similar sequence detailing currently available machines, whose examples are shown in Table 2. It is noted that the quantity and types of machines required for Model #1 were available in and thus fully covered by Production Line #2. Similarly, those for Model #2 are also covered by Production Line #7.

Table 2 Example models and production lines data

Machine Type	Model		Line	
	#1	#2	#2	#7
BAS-326H-0	2	4	5	4
BAS-311H	-	-	-	2
GC20618-2	2	3	2	3
S-7300A	3	-	3	-
BAS-342H	-	-	1	1
B-8452B-7	-	1	-	1
INVERT MACHINE	1	1	1	1
METAL DETECTOR	1	1	1	1
TABLE INSPECTION	2	1	2	1
TABLE SET FRONT	1	-	1	-
TEMPLATE BLACKLINE	-	-	1	-
AUTO STAMPING (LONG)	-	1	-	1
COMPUTER MONITOR	1	1	1	1
BARCODE PRINTER	2	2	2	2

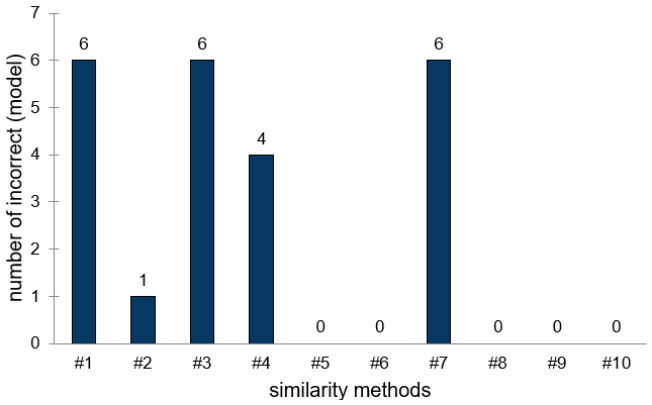
4. Preliminary Experiments

In the following experiments, the plant studied consisted of 10 lines producing 84 products. Each line may be able to produce more than one product model, and vice versa. Table 3 associates these production lines, their eligible products and counts (in parenthesis). It may be noted from the table that, for example, model #33 is produced by either line #2 or #3.

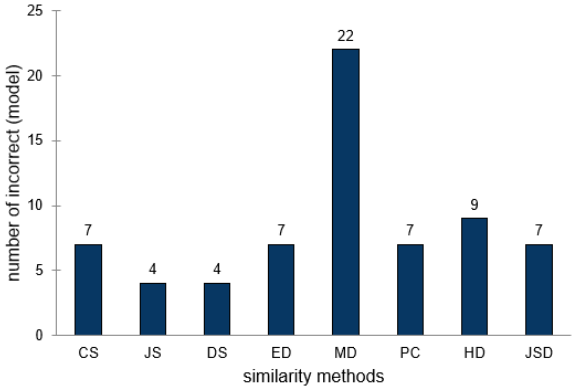
Table 3 Associations between production line and product model

Line ID	Models (Count)	Line ID	Models (Count)
1	1–13 (13)	6	58–62 (5)
2	14–24, 29, 33, 39, 40, 45, 46, 49, 53, 56, 57 (21)	7	36, 63–68, 81, 82, 84 (10)
3	24–33, 37 (11)	8	69–75 (7)
4	34–36, 38–43 (9)	9	76–79 (4)
5	32, 35, 39, 40, 44–57 (18)	10	80–84 (5)

In the first experiment, each product model was analyzed in turn. Its corresponding design sequence was matched against that of each product line, based on CS. The line with the highest CS was considered the most suitable for the model. The resultant line was then verified against its eligible models in Table 3, whether such model-line pair was indeed a correct association. The matching was repeated but using the remaining similarity metrics in turn (i.e., JS, ED, MD, PC, DS, HD, and JSD). The numbers of actual and incorrect associations were noted. For each model, successful case was identified only if every similarity metric gave the correctly matched line. Otherwise, it was marked as a failure. Out of 84 models, the number of failed cases found in each production line and those given by each similarity metric are shown in Figure 1(a) and 1(b), respectively. The total models successfully matched with eligible lines regardless of similarity were 61 cases, accounting for 72.62%.



(a)



(b)

Figure 1 Number of incorrect results found in each production line (a) and given by each similarity metric (b)

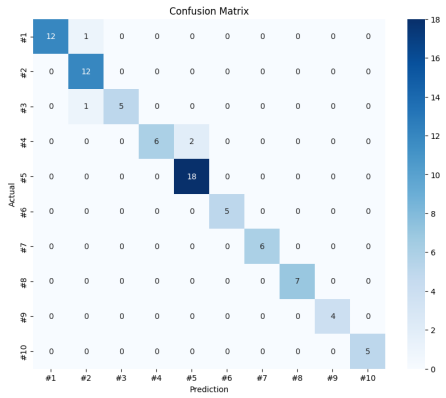
It is evident from Figure 1(b) that MD gave the highest error rate of 22 incorrect matching. On the contrary, both JS and DS performed equally well. They wrongly predicted the pairs for only 4 model instances, i.e., number 11, 21, 30 and 43. Therefore, these metrics gave the highest accuracy of about 95.24%.

Table 4 presents a comparative analysis of the variance and standard deviation (SD) of similarity scores across similarity matrices, revealing notable differences in result consistency. HD and JSD exhibited low variability, indicating highly consistent outcomes. Conversely, MD and PC showed higher variability, suggesting fewer stable results. Notably, the JS and DS demonstrated moderate variance and standard deviation (JS: Variance=0.0249, SD=0.1577, DS: Variance=0.0202, SD=0.1422), reflecting a balance between result stability and sensitivity to the differences.

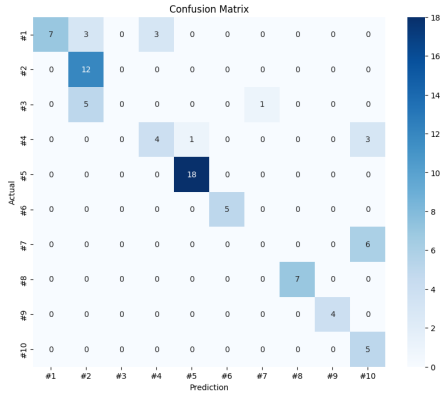
Table 4 Comparison SD and variance of similarity matrices

Similarity	Variance of Similarity	SD of Similarity
CS	0.01477	0.12155
JS	0.02488	0.15774
DS	0.02023	0.14222
ED	0.03521	0.18764
MD	0.84150	0.91733
PC	0.04506	0.21228
HD	0.00692	0.08321
JSD	0.00989	0.09944

Their balance makes JS and DS particularly suitable for structured data formats, as described in Section 3. Both similarity metrics emphasize the proportion of common elements through intersection operations, as shown in Eq. (2) and (3). Their set-based design makes them particularly well-suited for feature-level comparisons, where the presence or absence of elements holds greater significance than their sequential order or frequency. This characteristic contributes to their robustness and effectiveness in identifying appropriate model-production line associations in this research. Figure 2 compares the corresponding confusion matrices derived from JS and DS (a) and MD (b) predictions.



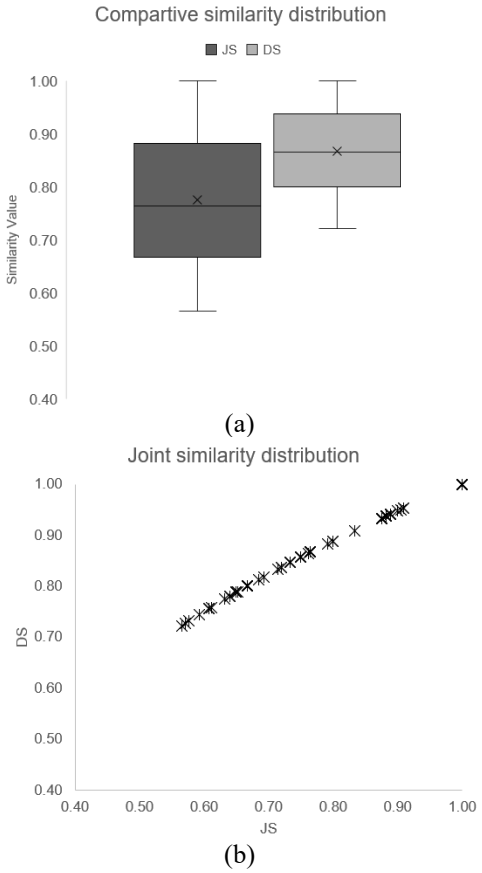
(a)



(b)

Figure 2 Confusion matrix resulted from JS and DS (a) and MD (b)

However, the actual similarity of these models, given by both metrics were different. For model #11, the JS and DS similarity were 0.565217 and 0.722222, respectively. Likewise, those for model #21 were 0.611111 and 0.758621, for model #30 were 0.714286 and 0.833333, and for model #43 were 0.65 and 0.787879, respectively. It can be observed that JS were typically lower than DS for the same model, but they were correlated. This observation also applies to the correctly predicted associations. Therefore, Figure 3 compares the individual (a) and multivariate (b) distributions between JS and DS across all models.



(a)

(b)

Figure 3 Box-whisker plot of JS and DS across all models (a) and (b) their joint distribution

Detailed analysis of individual models yields further insights into the involved similarity metrics. Take model #13 for example. Its similarity measures with respect to individual production lines are listed in Table 5. It can be read from the table that both JS and DS found line #1 the most appropriate match. Meanwhile, all the other remaining measures suggested line #2 instead. However, in reality, line #2 was not a viable choice, because it had only two GC20618-machines, whereas model #13 required three. Moreover, the line also lacked the B-8452B-7 and S-7200C machines required to produce this model.

Table 5 Similarity measures obtained from model #13 with respect to 10 production lines

Line	(1) CS	(2) JS	(3) DS	(4) ED	(5) MD	(6) PC	(7) HD	(8) JSD
#1	0.817568	**0.692308**	**0.818182**	0.604039	1.920981	0.543634	0.117647	0.301503
#2	**0.887012**	0.56	0.717949	**0.47537**	**1.002873**	**0.754024**	**0.088235**	**0.253065**
#3	0.819016	0.535714	0.697674	0.601637	2.052296	0.540635	0.176471	0.329703
#4	0.654768	0.407407	0.578947	0.830942	2.879903	0.21159	0.294118	0.450348
#5	0.755996	0.423077	0.594595	0.698575	2.337977	0.446078	0.205882	0.380229
#6	0.540853	0.37931	0.55	0.958277	3.836308	0.140463	0.264706	0.477375
#7	0.607052	0.52	0.684211	0.886508	3.276759	0.178281	0.323529	0.47655
#8	0.771747	0.346154	0.514286	0.675652	2.308089	0.537679	0.176471	0.365979
#9	0.513829	0.24	0.387097	0.986074	4.209822	0.135614	0.411765	0.563319
#10	0.554477	0.37037	0.540541	0.943953	3.642498	0.138968	0.333333	0.517191

Therefore, closer inspections on both production sequences were made to determine sources of prediction errors. Table 6 lists the sequences of model #13, line #1 and line #2. It is evident that line #2 had S-7300A, whereas model #13 demanded S-7200C. These elements differed by only two characters (positions), i.e., 3 versus 2 and A versus C. In addition, the words count in model 13 was more similar to line #2 than to line #1.

Table 6 The sequence required by model #13 compared to those available in lines #1 and #2

Machine Type	**Model**	**Line**	
	#13	**#1**	**#2**
BAS-326H-0	4	6	5
BAS-311H	-	-	-
GC20618-2	3	3	2
S-7200C	1	1	-
S-7300A	1	1	3
BAS-342H	-	-	1
B-8452B-7	1	2	-
INVERT MACHINE	1	1	1
METAL DETECTOR	1	1	1
TABLE INSPECTION	2	2	2
TABLE SET FRONT	1	1	1
TEMPLATE BLACKLINE	2	2	1
AUTO STAMPING(CIRCLE)	-	1	-
AUTO STAMPING(SUB)	-	1	-
AUTO STAMPING (LONG)	-	-	-
COMPUTER MONITOR	1	1	1
BARCODE PRINTER	2	2	2
Words Count	**12**	**14**	**11**

5. The Proposed Algorithm

It can be drawn from the above preliminary results that the words count within, and the length of the dataset played important roles in predictions. Whilst JS and DS outperformed the others and as such were the preferred choices, none of the metrics alone were adequate for this task. Therefore, this paper additionally introduces word segmentation and conditional priority assignments to the text data to enhance the prediction accuracy. They are outlined in Table 7.

Table 7 The proposed priority assignment scheme

Priority	Description
1	The machine types for the model must be fully included in the production line.
2	The number of machines in the production line must be no less than that required by the model.

The proposed system begins with database preparation. The database consisted of production line and product model textual data, as outlined in Section 3. These data were then passed to the main algorithm, where their similarity measures were calculated, and the proposed heuristically conditional processes were taken. Detailed description of the algorithm is given in the next section. The resulting prediction, i.e., a production line that was best suited for the given (unseen) model, was finally reported. The diagram describing the proposed line prediction system is shown in Figure 4.

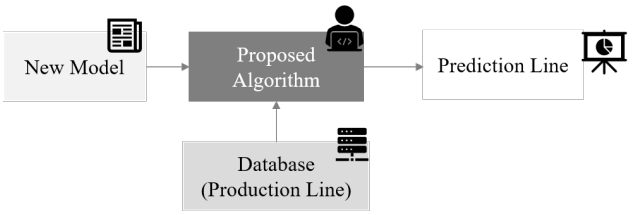


Figure 4 Overview of the proposed system

The process of the proposed method was divided into three main steps:

(1) *Word Segmentation*

Firstly, a given textual sequence was segmented by the comma (,) delimiter. A condition was then imposed on an eligible pair, i.e., the words count in the product model must be less than or equal to that in the production line. The condition can be expressed by $B \subseteq A$, where A and B were set of elements in the line and model, respectively. Only the segmented machine types were considered when measuring the similarities.

(2) *Ensemble Similarity*

For a given line and model pair, JS and DS were computed. For the pair to be admissible to the prediction results, its JS and DS must be no less than 0.75 or 0.85, respectively. The thresholds were determined from the preliminaries reported in Figure 3.

(3) *Conditional Priority Assignments*

Cases failing to reach the above JS and DS thresholds were compared. Unlike the previous step, machine types from the production line that were not present in the model were removed before the comparison. Then, JS and DS were recomputed for the filtered sequences, consisting of the remaining machine types. If similarity values were 1.0, the line was admissible to the prediction only if it also satisfied the condition specified in (1), i.e., the number of each machine type in the model must be no more than that available on the production line.

The above process is summarized in a flowchart as depicted in Figure 5.

6. Results and Discussion

For the 84 product models and 10 production lines involved in this study. There were 840 cases (i.e., model-line pairs) to be considered in total.

(1) *Word Segmentation*

Out of the 840 cases, there were only 654 cases that met the criterion for the word counts set in Section 5 (1). In other words, approximately 23% of the total cases were excluded in this step. Presented in Figure 6 are the numbers of cases (i.e., production lines) satisfying this condition for each model. It can be noted that initially there were about 3 to 10 candidate lines for a given model.

(2) *Ensemble Similarity*

Subsequently, JS and DS for the remaining 645 cases were computed. Within each model, the cases were ranked based on these values in descending order. Cases where the JS and DS did not meet the referent thresholds set in the second step were excluded. The higher the similarity, the greater the resemblance within the same group (model). As can be seen in Figure 7, there are only 48 out of 84 models, whose at least one case passed this criterion. To demonstrate some eligible cases, Figure 8 lists some final predictions, e.g., lines #2 and #7 were the most suitable (i.e., with the highest similarity) for models #14 and #65, respectively. The remaining 36 models thus required further assessment in the final step.

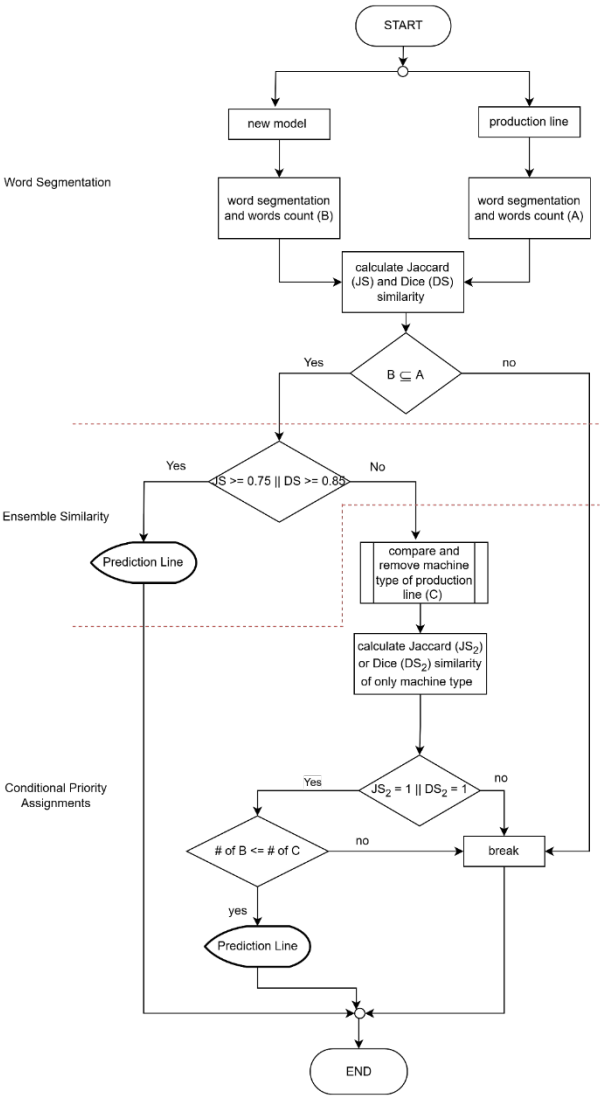


Figure 5 Flowchart of proposed prediction system.

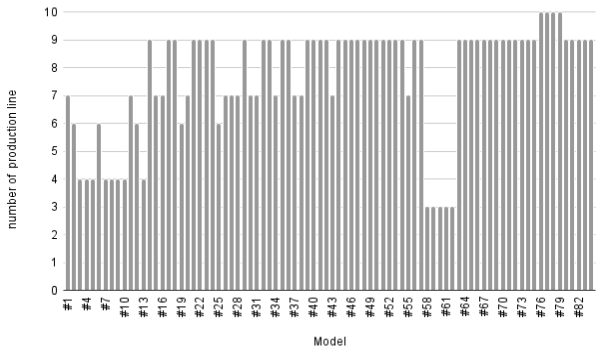


Figure 6 The number of production lines that satisfied the condition in the first step (1)

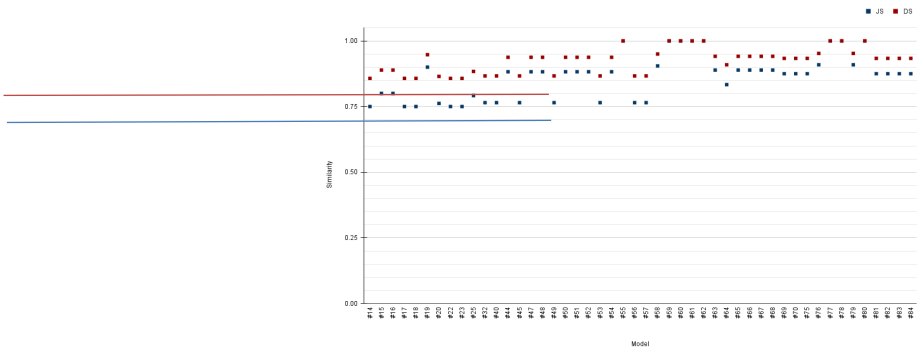


Figure 7 JS and DS similarity for all 84 cases and those reaching the threshold set in the second step (2)

```
Number of Test Cases passed the condition: 48
Test Cases that passed the condition: ['Model #14: Line #2', 'Model #15: Line #2', 'Model #16: Line #2', 'Model
#17: Line #2', 'Model #18: Line #2', 'Model #19: Line #2', 'Model #20: Line #2', 'Model #22: Line #2', 'Model
#23: Line #2', 'Model #25: Line #3', 'Model #32: Line #5', 'Model #40: Line #5', 'Model #44: Line #5', 'Model
#45: Line #5', 'Model #47: Line #5', 'Model #48: Line #5', 'Model #49: Line #5', 'Model #50: Line #5', 'Model
#51: Line #5', 'Model #52: Line #5', 'Model #53: Line #5', 'Model #54: Line #5', 'Model #55: Line #5', 'Model
#56: Line #5', 'Model #57: Line #5', 'Model #58: Line #6', 'Model #59: Line #6', 'Model #60: Line #6', 'Model
#61: Line #6', 'Model #62: Line #6', 'Model #63: Line #7', 'Model #64: Line #7', 'Model #65: Line #7', 'Model
#66: Line #7', 'Model #67: Line #7', 'Model #68: Line #7', 'Model #69: Line #8', 'Model #70: Line #8', 'Model
#75: Line #8', 'Model #76: Line #9', 'Model #77: Line #9', 'Model #78: Line #9', 'Model #79: Line #9', 'Model
#80: Line #10', 'Model #81: Line #10', 'Model #82: Line #10', 'Model #83: Line #10', 'Model #84: Line #10']
```

Figure 8 Examples of cases whose production lines satisfied the similarity value condition.

(3) *Conditional Priority Assignments*

Finally, the prediction results for the remaining 36 models and their associated cases that satisfied the last condition are shown in Figure 9. Since the suitable production lines for a given models were known (as shown in Table 4), their confusion matrix can be determined from the resultant prediction and shown in Figure 10. It can be noted that all production lines were predicted correctly. The accuracy of the proposed method was therefore 100%.

```
['Model #1: Line #1', 'Model #2: Line #1', 'Model #3: Line #1', 'Model #4: Line #1', 'Model #5: Line #1', 'Model #6: Line #1',
'Model #7: Line #1', 'Model #8: Line #1', 'Model #9: Line #1', 'Model #10: Line #1', 'Model #11: Line #1', 'Model #12: Line
#1', 'Model #13: Line #1', 'Model #21: Line #4', 'Model #21: Line #2', 'Model #21: Line #3', 'Model #21: Line #1', 'Model #24:
Line #2', 'Model #24: Line #3', 'Model #24: Line #1', 'Model #26: Line #3', 'Model #27: Line #3', 'Model #28: Line #3', 'Model
#29: Line #2', 'Model #29: Line #3', 'Model #29: Line #1', 'Model #30: Line #3', 'Model #31: Line #3', 'Model #33: Line #2',
'Model #33: Line #3', 'Model #33: Line #1', 'Model #34: Line #4', 'Model #35: Line #5', 'Model #35: Line #4', 'Model #35: Line
#2', 'Model #35: Line #3', 'Model #35: Line #1', 'Model #36: Line #4', 'Model #36: Line #3', 'Model #37: Line #4', 'Model #37:
Line #3', 'Model #38: Line #4', 'Model #39: Line #5', 'Model #39: Line #4', 'Model #39: Line #2', 'Model #39: Line #3', 'Model
#39: Line #1', 'Model #41: Line #4', 'Model #42: Line #4', 'Model #43: Line #4', 'Model #43: Line #1', 'Model #46: Line #5',
'Model #46: Line #4', 'Model #46: Line #2', 'Model #46: Line #3', 'Model #71: Line #8', 'Model #72: Line #8', 'Model #73: Line
#8', 'Model #74: Line #8']
```

Figure 9 Examples of cases whose production lines satisfied the final condition (3)

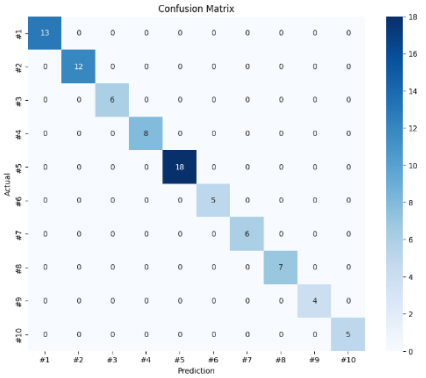


Figure 10 Confusion matrix of proposed algorithm

In our experiment, some product models can be produced by more than one production line. For example, model #21 can be produced by lines #4, #2, #3, or #1. They were ranked by their similarity measures.

To demonstrate the merits of the proposed method, the red box in Figure 9 highlights three models initially misclassified in the initial prediction. However, the proposed algorithm finally enabled accurate associations for these production lines. For example, as shown in Table 8, production line #5 gave the highest similarity value for model #21. However, this line had only two BAS-326H-0 machines. The number of this machine is insufficient, because the designated model required three. Therefore, line #5 cannot be used to produce model #21.

With the proposed algorithm, the process is divided into three steps. (1) Word Segmentation: This step verifies whether the segmented words from the model are a subset of those from the production line. In this case, all three lines satisfy this condition. (2) Ensemble Similarity: The JS and DS values between model #21 and the data from lines #5, #4, and #2 (which are the top three candidates with the highest JS and DS scores) are shown in Table 9. It can be observed that JS < 0.75 and DS < 0.85; therefore, the process proceeds to (3) Conditional Priority Assignments. In this step, filtering of the candidate lines is performed first (as shown in Table 8), and then JS and DS are recalculated. The results show that both JS and DS reach the value of 1.0. Finally, the number of machines required by model #21 is compared with those available in the filtered lines. Upon completion of the proposed algorithm, lines #4 and #2 satisfied the last condition. They contained four and five BAS-326H-0 machines, respectively. The number exceeded that needed by the model.

Table 8 The sequence required by model #21 compared to those available in lines #5, #4 and #2

Machine Type	Model	Line			Filtering Line		
	#21	#5	#4	#2	#5	#4	#2
BAS-326H-0	3	2	4	5	2	4	5
BAS-311H	-	-	2	-	-	-	-
GC20618-2	2	2	2	2	2	2	2
S-7300A	-	2	2	3	-	-	-
BAS-342H	-	-	-	1	-	-	-
B-8452B-7	-	-	2	-	-	-	-
PQ1500SL	-	-	2	-		-	-
INVERT MACHINE	1	1	1	1	1	1	1
METAL DETECTOR	1	1	1	1	1	1	1
TABLE INSPECTION	2	2	2	2	2	2	2
TABLE SET FRONT	-	-	-	1	-	-	-
TEMPLATE BLACKLINE	-	1	-	1	-	-	-
AUTO STAMPING(CIRCLE)	-	1	1	-	-	-	-
COMPUTER MONITOR	1	1	1	1	1	1	1
BARCODE PRINTER	2	2	2	2	2	2	2
Words Count	**7**	**10**	**12**	**11**	**7**	**7**	**7**

Table 9 The initial prediction results (line #5) and those obtained by the proposed method (lines #4 and #2) for model #21

Line	Ensemble Similarity		Conditional Priority Assignments		Result
	JS	DS	JS	DS	
#5	0.6111	0.7586	1	1	✗
#4	0.5789	0.7333	1	1	✓
#2	0.5500	0.7097	1	1	✓

7. Conclusions

This paper proposes a production line prediction method. It aimed at enhancing planner's efficiency, particularly following the design extraction phase of the product drawing. The proposed algorithm utilized a two-step similarity measure, comparing two textual datasets, using Jaccard and Dice similarity (JS and DS). A notable advantage of this similarity-based approach is that it is consistent across different letter cases, producing identical results for both uppercase and lowercase encoding texts. In the experiments conducted on 84 models, a single-step application of JS and DS achieved an accuracy of only 95.24%. With the proposed two-step similarity approach, accuracy of 100% could be obtained. Our findings indicate that the proposed method can be applied to predict a new product model unseen to the manufacturing plant.

In future work, focus should be aimed at enhancing the efficiency of the proposed algorithm by enabling it not only to predict the most suitable production line, but also to provide specific recommendations regarding the selected alternative line. This includes identifying any missing machines within that production line and specifying both the quantity and types of equipment required. Such improvements would support practical decision-making in manufacturing environments where resources may vary, and ensure a more comprehensive deployment of the algorithm in real-world applications.

Acknowledgements

This work is financially supported by the Faculty of Engineering, Burapha University (Grant number 1/2568).

References

[1] N. Liyanaarachchi, J. Weng, and S. Akasaka, "A Review of Literature on Engineer-To-Order Production systems," *Asian Journal of Management Science and Applications*, vol. 8, no. 1, pp. 53-82, 2023, do: 10.1504/AJMSA.2023.134445.

[2] S. Jia, C. Wang, X. Qing, and Z. Ma, "Organization Design of Production Line for an Enterprise," *8th International Conference on Industrial Engineering and Applications (ICIEA)*, Chengdu, China, 2021, pp. 252-256, doi: 10.1109/ICIEA52957.2021.9436764.

[3] M. Mallampati, K. Srivinivas, and T. Krishna. M, "Design Process to Reduce Production Cycle Time in Product Development," *IAES International Journal of Artificial Intelligence*, vol. 7, no. 3, pp. 125-129, 2018, doi: 10.11591/ijai.v7.i3.pp125-129.

[4] M. R. Prajapati and V. A. Deshpande, "Cycle Time Reduction using Lean Principles and Techniques: A Review," *International Journal of Advance Industrial Engineering*, vol. 3, no. 4, pp. 208-213, 2015.

[5] N. B. Kacar, L. Mönch, and R. Uzsoy, "Problem Reduction Approaches for Production Planning using Clearing Functions," *14th International Conference on Automation Science and Engineering (CASE)*, Munich, Germany, 2018, pp. 931-938, doi: 10.1109/COASE.2018.8560429.

[6] N. C. Nwasuka and U. Nwaiwu, "Computer-based Production Planning, Scheduling and Control: A Review," *Journal of Engineering Research*, vol. 12, no. 1, pp. 275-280, 2024, doi: 10.1016/j.jer.2023.09.027.

[7] V. Modrak, R. Sudhakarapandian, A. Balamurugan, and Z. Soltysova, "A Review on Reinforcement Learning in Production Scheduling: An Inferential Perspective," *Algorithms*, vol. 17, no. 8, pp. 343-343, 2024, doi: 10.3390/a17080343.

[8] R. Bandinelli and V. Fani, "Combined use of AI Techniques and Simulation to Support Production Scheduling: evidence from Empirical Research," *38th ECMS International Conference on Modelling and Simulation*; 2024. pp. 34-40, doi: 10.7148/2024-0034.

[9] L. Wang, H. Liu, M. Xia, Y. Wang, and M. Li, "Research on a Multilevel Scheduling Model for Multi Variety and Variable batch Production Environments based on Machine Learning," *Frontiers in Energy Research*, vol. 11, 2023, doi: 10.3389/fenrg.2023.1251335.

[10] N. C. Nwasuka and U. Nwaiwu, "Computer-based Production Planning, Scheduling and Control: A Review," *Journal of Engineering Research*, vol. 12, no. 1, pp. 275-280, 2024, doi: 10.1016/j.jer.2023.09.027.

[11] L. Wang, H. Liu, M. Xia, Y. Wang, and M. Li, "A Machine Learning based EMA-DCPM Algorithm for Production Scheduling," *Scientific Reports*, vol. 14, no. 20810, 2024, doi: 10.1038/s41598-024-71355-w.

[12] X. Ding and B. Zheng, "Research on Intelligent Planning Algorithms based on Machine Learning Optimization," *International Conference on Computers, Information Processing and Advanced Education (CIPAE)*, 2023, pp. 94-99, doi: 10.1109/CIPAE60493.2023.00024.

[13] Z. Luo, C. Jiang, L. Liu, X. Zheng, H. Ma, and F. Dong, "Deep Reinforcement Learning based Production Scheduling in Industrial Internet of Things," *IEEE Internet of Things Journal*, vol. 10, no. 22, pp. 19725-19739, 2023, doi: 10.1109/JIOT.2023.3283056.

[14] H. Song, I. Gi, J. Ryu, Y. Kwon, and J. Jeong, "Production Planning Forecasting System Based on M5P Algorithms and Master Data in Manufacturing Processes," *Applied Sciences*. 2023; vol. 13, no. 13, pp. 7829, doi: 10.3390/app13137829.

[15] C. Yuan, C.-C. Wang, M.-L. Chang, W.-T. Lin, P.-A. Lin, C.-C. Lee, and Z.-L. Tsui, "Using a Flexible IoT Architecture and Sequential AI Model to Recognize and Predict the Production Activities in the Labor-Intensive Manufacturing Site," *Electronics*, vol. 10, no. 20, pp. 2540, 2021, doi: 10.3390/electronics10202540.

[16] A. P. Alodjants, A. E. Avdyushina, D. V. Tsarev, I. A. Bessmertny, and A. Y. Khrennikov, "Quantum Approach for Contextual Search, Retrieval, and Ranking of Classical Information," *Entropy*, vol. 26, no. 10, pp. 862, 2024, doi: 10.3390/e26100862.

[17] AN. Surya and J. Vimala, "Similarity Measure for Complex Non-Linear Diophantine Fuzzy Hypersoft Set and its Application in Pattern Recognition," *Information Sciences*, vol. 690, pp. 121591, 2025, doi: 10.1016/j.ins.2024.121591.

[18] M. Azam, Y. Chen, M. O. Arowolo, H. Liu, M. Popescu, and D. Xu, "A Comprehensive Evaluation of Large Language Models in Mining Gene Relations and Pathway Knowledge," *Quantitative Biology,* vol. 12, no. 4, pp. 360-374, 2024, doi: 10.1002/qub2.57.

[19] C. Sun, G. Che, X. Dong, R. Zou, L. Feng, and X. Ding, "Review on Algorithm for Fusion of Oblique Data and Radar Point Cloud," *Lecture Notes in Electrical Engineering*. vol. 1033, pp. 527-535, 2024, doi: 10.1007/978-981-99-7502-0_58.

[20] I. Bartkowska, L. Wysocki, A. Zajkowski, and P. Tuz, "Comparative Analysis of Leak Detection Methods using Hydraulic Modelling and Sensitivity Analysis in Rural and Urban–Rural Areas," *Sustainability*, vol. 16, no. 17, 2024, doi: 10.3390/su16177405.

[21] S. Shumet, E. Salelew, G. T. Desalegn, Y. Mirkena, D. A. Angaw, T. A. Zeleke, T. Kassew, and M. Wondie, "Socio-Demographic, Psychosocial, and Suicidal Behavior correlates of Stigma Among People with Physical Disabilities in Northwest Ethiopia," *BMC Public Health*, vol. 24, 2024, doi: 10.1186/s12889-024-20379-y.

[22] K. Pani and I. Chawla, "Synthetic MRI in Action: A Novel Framework in Data Augmentation Strategies for Robust Multi-Modal Brain Tumor Segmentation," *Computers in Biology and Medicine*, vol.183, 2024, doi: 10.1016/j.compbiomed.2024.109273.

[23] H. Arabi and M. Akbari, "Improving Plagiarism Detection in Text Document using Hybrid Weighted Similarity," *Expert Systems with Applications*, vol. 207, 2022, doi: 10.1016/j.eswa.2022.118034.

[24] X. Wang and E.V. Konstantinova, "The Sequence Reconstruction Problem for Permutations with the Hamming Distance," *Cryptography and Communications*, vol. 16, pp. 1033-1057, 2024, doi: 10.1007/s12095-024-00717-y.

[25] F. Shao, H. Shao, D. Wang, and W.H.K. Lam, "A Multi-task Spatio-Temporal Generative Adversarial Network for Prediction of Travel Time Reliability in Peak Hour Periods," *Physica A: Statistical Mechanics and its Applications*, vol. 638, 2024, doi: 10.1016/j.physa.2024.129632.