


Article

Ensemble Meta-Learning-Based Robust Chipping Prediction for Wafer Dicing

Bao Rong Chang ¹, Hsiu-Fen Tsai ^{2,*} and Hsiang-Yu Mo ¹

¹ Department of Computer Science and Information Engineering, National University of Kaohsiung, Kaohsiung 81148, Taiwan; brchang@nuk.edu.tw (B.R.C.); m1105519@mail.nuk.edu.tw (H.-Y.M.)

² Department of Fragrance and Cosmetic Science, Kaohsiung Medical University, Kaohsiung 80708, Taiwan

* Correspondence: sftsai@kmu.edu.tw

Abstract: Our previous study utilized importance analysis, random forest, and Barnes–Hut t-SNE dimensionality reduction to analyze critical dicing parameters and used bidirectional long short-term memory (BLSTM) to predict wafer chipping occurrence successfully in a single dicing machine. However, each dicing machine of the same type may produce unevenly distributed non-IID dicing signals, which may lead to the undesirable result that a pre-trained model trained by dicing machine #1 could not effectively predict chipping occurrence in dicing machine #2. Therefore, regarding the model robustness, this study introduces an ensemble meta-learning-based model that can evaluate many dicing machines for chipping prediction with high stability and accuracy. This approach constructs several base learners, such as the hidden Markov model (HMM), the variational autoencoder (VAE), and BLSTM, to form an ensemble learning. We use model-agnostic meta-learning (MAML) to train and test the ensemble learning model by several prediction tasks from machine #1. After MAML learning, we call the trained model a meta learner. Then, we successfully apply a retrieved data set from machine #2 to the meta learner for training and testing wafer chipping occurrence in this machine. As a result, our contribution to the robust chipping prediction on cross-machines can improve the yield of wafer dicing with a prediction accuracy of 93.21%, preserve the practical wearing of dicing kerfs, and significantly cut wafer manufacturing costs.

Keywords: wafer dicing; robust chipping prediction; random forest; dimensionality reduction; model-agnostic meta-learning; ensemble meta-learning



Citation: Chang, B.R.; Tsai, H.-F.; Mo, H.-Y. Ensemble Meta-Learning-Based Robust Chipping Prediction for Wafer Dicing. *Electronics* **2024**, *13*, 1802. <https://doi.org/10.3390/electronics13101802>

Academic Editors: Lien Minh Dang and Hyeonjoon Moon

Received: 6 March 2024

Revised: 25 April 2024

Accepted: 3 May 2024

Published: 7 May 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the advent of the AI boom, competition and cooperation in advanced wafer foundries will affect the prospects of various semiconductor companies and the new development process of the global semiconductor industry. System wafer foundries designed for AI vary from technology competitions to international geopolitical arrangements, especially cutting-edge process chips, which are crucial to the development of AI. There is a need to improve the production of AI chips and, more importantly, to integrate AI technology into high-end semiconductor processes to improve production yield and reduce manufacturing losses. Texas Instruments, in 1993, cooperated with the United States military and conducted advanced process control (APC) or advanced equipment control (AEC) [1,2] instead of statistical process control (SPC). These process control methods have become quite mature technologies. To further reduce wafer manufacturing costs, wafer fabs are considering whether APC can improve process control capabilities and equipment usage efficiency to optimize the process and increase capacity utilization. Fabs with AI approaches, for example, can effectively control the occurrence of the backside wall chipping during wafer dicing. The main topic of this paper is how to prevent the occurrence of chipping in wafer dicing.

In our previous study [3], we proposed several methods to check dicing signals generated from the machine during wafer dicing that can analyze fab machines to see

any precursors or common denominators for wafer chipping and take measures to avoid things happening. The first was to examine the common correlation using Spearman [4–6], Pearson [7,8], and Kendall [9] methods. The previous work chose one to analyze the log data according to the characteristics of data with/without linearity and continuity. In this way, the correlation analysis gave information about a correlation between different parameters in the log, looking at combining other machine parameters. For instance, wafer chipping can appear due to various factors in wafer dicing, such as kerf wear, cooling water temperature, and cleaning gas emissions. Next, further information analysis, like importance analysis [10], can extract the important dicing parameters that affect wafer chipping. Finally, random forest [11,12] examined the judgment nodes to screen the critical dicing parameters from the important ones. Consequently, we can see the changes in wafer chipping occurrence to understand the critical parameters verified by correlation analysis [13].

Precise and complex dicing machines can be suitable for controlling advanced manufacturing processes to avoid a large-scale area of wafer chipping. In other words, the production machines may generate hundreds or even thousands of control parameters during wafer dicing. Constructing the detection or prediction model using many high-dimensional parameter vectors is tough for wafer dicing. If a slight deviation in the parameter value happens, it could cause detection or prediction deviation, leading to a significant error in the model output. Therefore, our previous work [3] implemented the dimensionality reduction of any high-dimensional parameter vector to a single one-dimensional indicator with PCA [14,15] or Barnes–Hut t-SNE [16,17]. After that, we established a bidirectional long short-term memory (BLSTM) [18,19] to predict wafer chipping occurrence. As a result, this model can work successfully in chipping occurrence prediction as per a pre-specified wafer dicing machine.

Many early dicing machines, e.g., the DFD 6560, are still available in wafer dicing operations in fabs. Judging the changes in dicing signals for promptly tuning machine parameters can reduce the occurrence of large-area wafer chipping. Our previous study [3] made a BLSTM to precisely predict chipping occurrence in a single specified dicing machine, e.g., machine #1. However, we applied transfer learning to the chipping prediction on machine #2 from the pre-trained model in machine #1, and it turned out to be the worst situation, significantly lowering the chipping prediction accuracy on machine #2. The pre-trained BLSTM lacked the robustness of chipping prediction in wafer dicing. The problem is enhancing the robustness of chipping prediction on cross-machines to maintain high prediction accuracy. This study proposes model-agnostic meta-learning [20] to breed a pre-learned base learner called a meta learner that can tackle a new task or environment in the future. However, another problem is that a specified model may not cope with the varying characteristics of cross-machines. Again, this study will establish an ensemble learning [21,22] with multiple base learners, including the hidden Markov model (HMM) [23,24], variational autoencoder (VAE) [25,26], and BLSTM. Such a combination can deal with the varying characteristics of cross-machines and improve the overall system performance to achieve the best prediction. The contribution of this study is to propose a new chipping prediction framework called ensemble meta-learning that can adapt the new task or environment between machines to appropriately adjust the critical parameters of dicing machines to avoid large-scale chipping occurrence, effectively improving production yield and reducing manufacturing losses.

2. Literature Review and Background Material

2.1. Literature Review

In the recent development of Industry Revolution 4.0, big data analytics have been applied to large amounts of data processing and analyses [27]. In addition to the correlation analysis to know the data relationship [13], time series technology can observe and predict the data sequence based on a timeline [28]. Regarding intelligent computing, the model using the machine learning method can learn prior data to infer the predictive outcome

shortly, and its applications become increasingly important. Random forest [11,12] is one of the most common approaches in machine learning for data analysis and decision making. People can usually use importance analysis [10] to discover a few important factors out of thousands or hundreds of factors in a system. The random forest method is relatively simple and transparent, making it more straightforward to see the judgment of different parameters at each node during operation. It helps quickly screen a few critical factors from the important ones.

During IC assembly, testing, and packaging (ATP), the machine will test again through the wafer surface after wafer dicing. This chipping examination can prevent severely defective chips from going directly to assembly, resulting in many poorly assembled ICs. How to detect or predict the possible chipping occurrence in advance during wafer dicing becomes a more critical issue. Chipping detection technology has been widely used in the ATP process by semiconductor factories. Li et al. [29] stated that we can train several models before detecting wafer surface chipping phenomena. Tsuda et al. [30] also mentioned that many online databases provide a large amount of data about the chipping phenomena to anyone who wants to use them for modeling in the training and testing phases.

Furthermore, people are concerned about reducing the manufacturing loss caused by chipping if the method incorporated with the FDC system can prevent wafer damage arising in the ATP process. Although most of the recent literature, e.g., Fan et al. proposed in [31] and Sunny et al. introduced in [32] applied machine-generated data to simple statistics or machine learning for modeling, where they cannot achieve the required accuracy in the chipping occurrence prediction during wafer dicing. Nevertheless, Yang, S. [33] described that both LSTM and bidirectional LSTM models had higher accuracy in predicting a specific time series. According to the performance comparison between the two approaches, the prediction accuracy using the bidirectional LSTM model was better than the LSTM. Bidirectional LSTM models to predict abnormal data are also successful in high accuracy, verified by Liu et al. in [34].

Additionally, F. Gao et al. [23] mentioned the hidden Markov model (HMM), which reflects the stochastic behavior of the machine, reveals its hidden states, and changes the scheduled processes. For machines, the nature with comprehensive understanding can facilitate the estimate of changes in the status and performance in the prediction and assessment of nonlinear weak signals. A. Gong et al. [24] explained that HMM can evaluate the signals of fault prediction and health condition in machinery to assign the emerging areas with such signals. In other words, this action gives accurate predictions or evaluations of machine anomalies. Y. Zhao et al. [25] introduced variational autoencoder (VAE) to time series anomaly detection. VAE delivers a powerful probabilistic modeling framework for time series data modeling and analysis. With the probabilistic modeling capability, VAE can better understand the true distribution of given data to boost anomaly detection accuracy. T.-H. Kim et al. [26] described VAE performing anomaly detection well using its reconstructed data. Compared with the original data, reconstructed data for VAE make time series anomaly detection highly applicable.

Intelligent computing usually simplifies complex data to improve execution performance. Thus, it can effectively perform reasoning applications with better results. In other words, we can reduce the dimensionality of high-dimensional vectors to obtain low-dimensional vectors and simplify the high-dimensional complex data. Ou et al. [35] concluded that the proposed new PCA can beat the linear PCA method in dimensionality reduction. New PCA can optimize the reduction effect by using threshold filtering features and entropy. Deng et al. [36] introduced a nonlinear dimensionality reduction, t-SNE, which is good at classification tasks suitable for various nonlinear data sets. Yumeng et al. [37] mentioned that t-SNE is not good at big data analytics. Thus, they proposed an enhanced version, Barnes–Hut t-SNE, which is helpful for nonlinear big data analytics.

2.2. Time Series Anomaly Detection

Finding anomalies hidden in time series is a big problem; people usually use time series anomaly detection. Whether the system predicts abnormal phenomena as usual or vice versa, it will cause considerable losses in the time series applications. In addition, the time series is highly complex and changeable, so many normal or abnormal states may occur within a certain period, or abnormal states may scatter among normal states. This challenging situation often makes anomaly detection difficult. Adjusting the model sensitivity for anomaly detection is a huge challenge. Three methods can detect anomalies in time series as follows.

- (1) **Statistical approach** It assumes that the target data are of normal distribution. When the observed data exceed three times the standard deviation, statistics determine them to be abnormal (a simple and intuitive approach). However, it will fail when the data distribution is not normal. This method cannot initially identify spatial anomalies if the data contain high-dimensional data points.
- (2) **Supervised learning** We can apply a deep learning approach to anomaly detection by finding feature values from the data sequence and classifying the anomalies. However, supervised learning is the most time-consuming and labor-intensive because it relies on many manually labeled data to train the model. Still, it is usually more accurate than other learning methods. In addition, if the ratio of the number of normal data to abnormal data is seriously unbalanced, it will probably lead to poor performance of the trained classifier.
- (3) **Semi-supervised learning** The amount of data we collect is often too sparse, resulting in poor training results. Therefore, for example, we can use a simple three-hidden-layer autoencoder to learn the characteristics of normal distribution in data, as shown in Figure 1. Thus, the autoencoder can generate more training data and combine and restore the data with apparent errors that we identify as abnormal. Even though the autoencoder can achieve significant results in many fields, it will lead to low accuracy if the environment changes too quickly.

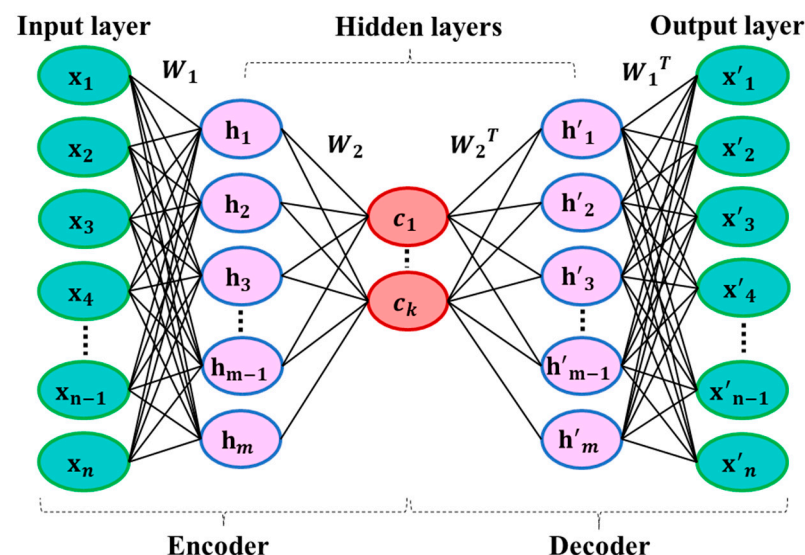


Figure 1. A simple three-hidden-layer autoencoder.

2.3. Data Preprocess

In wafer dicing, people encounter severe problems, such as loss of collected data, nonlinear data distribution, and loss of related hidden parameters. Our previous work [3] found that data sets collected from the wafer dicing process can reveal new information (hidden features) about the wafer representation, especially in wafer coverage areas smaller than 30 for a single wafer. In machine #1, the coverage area of wafer chipping ratio of

10%, 10~15%, 15~20%, and 20~30% (i.e., four groups of data sets) is equal to the number of samples in the single kerf dicing process from the beginning to the replacement of the wafer. Mainly, average pooling is applied to four group data sets by averaging four sampled data at the same corresponding position in each group to generate extra data that can deliver hidden information between data and increase the training data, as shown in Figure 2.

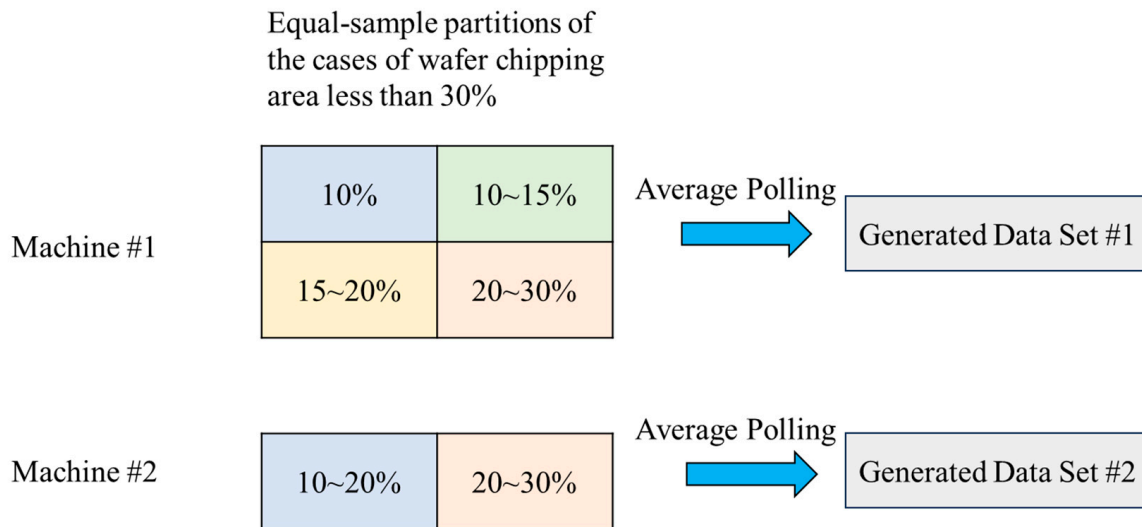


Figure 2. Generated data set using average pooling from individual machines.

Similarly, in machine #2, we find that the coverage area of wafer chipping ratio of 10~20% and 20~30% (i.e., two groups of data sets) is equal to the number of samples. Then, in Figure 2, the pooling average is applied to obtain extra data that can give hidden information between data and increase the amount of training data. Follow-up experiments will demonstrate the technical contribution and improve the prediction accuracy of chipping occurrence.

2.4. Data Dimensionality Reduction

Most dicing signals generated during wafer dicing have little influence on the chipping occurrence. Therefore, importance analysis [10] can effectively screen the relatively important dicing parameters for analyzing chipping phenomena afterward. However, prediction models built with high-dimensional input sample vectors may not achieve high prediction accuracy regarding the probability of chipping occurrence. Reduce implementing the dimensionality reduction of high-dimensional input sample vectors to a simple condense indicator as the input signal can benefit the modeling, with high prediction accuracy in this case.

Our previous study [3] proposed dimensionality reduction with linear PCA [14,15] and nonlinear t-SNE [16,17]. Two of them are shown below. The conditional probabilities $p_{j|i}$ in Equation (1) present the similarity between two high-dimensional sample points in a Gaussian distribution using t-SNE, where σ represents the variance, x_i stands for the current data, x_j indicates the following data of x_i , and X denotes a set of high-dimensional data. Equation (2) gives the probability density function (PDF) p_{ij} , where N represents the total amount of data between i and j . For low-dimensional data, Equation (3) computes the probability density function, q_{ij} , in the conditional probability of t distribution, where y_i denotes the current data, y_j indicates the following data of y_i , and Y denotes a set of low-dimensional data. Instead of Gaussian distribution, t distribution was applied to the low-dimensional vector to avoid a situation where the outliers would significantly affect the prediction result due to such information diminishing in the condensed indicator after dimensionality reduction. According to the low-dimensional data in the t distribution, we compute KL divergence to attain the loss function c in Equation (4) and then obtain the

gradient descent, $\frac{\delta C}{\delta y_i}$, in Equation (5). Thus, we can continuously update low-dimensional data using the derivative.

$$p_{j|i} = \frac{\exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma^2}\right)}{\sum_{k \neq i} \frac{\exp\left(\frac{-\|x_i - x_k\|^2}{2\sigma^2}\right)}{2\sigma^2}} \tag{1}$$

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N} \tag{2}$$

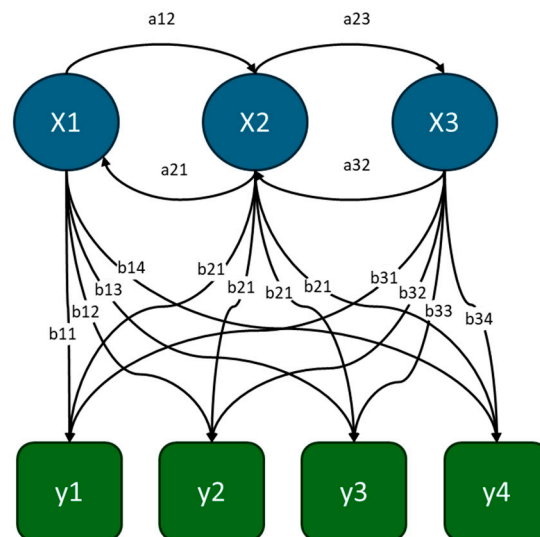
$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq i} (1 + \|y_k - y_i\|^2)^{-1}} \tag{3}$$

$$c = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{4}$$

$$\frac{\delta c}{\delta y_i} = 4 \cdot \sum_j (p_{ij} - q_{ij})(y_i - y_j)(1 + \|y_i - y_j\|)^{-1} \tag{5}$$

2.5. Hidden Markov Model (HMM) and Variational Autoencoder (VAE)

The Markov chain is the probability of the same type of events (different states) occurring sequentially because the state that occurs before will affect the state that arises later. It is a mathematical model used to infer this relationship, and the hidden Markov model is the flow chart of a model that predicts results by finding some hidden influencing factors, as shown in Figure 3. $x(t)$ is a hidden state or hidden variable. The observer cannot directly observe the hidden variable, so this is what we imagined, which means that some decision-making factors affected our results. $y(t)$ is the observation state or observation variable, which is what we observed. For example, if we toss a coin four times in a row, the results are positive, negative, negative, and positive. These four times are the states we observe, and each toss, the strength and direction of the hand when holding a coin, the wind speed of the air, etc., are hidden.



- X – states
- y – possible observations
- a – state transition probabilities
- b – output probabilities

Figure 3. HMM model.

The autoencoder uses the deep learning network to train the entire model through dimensionality reduction (encoder) and dimensionality enhancement (decoder). Finding the key dimensions makes it possible to reach the input and output patterns as closely as possible. The simple autoencoder still has some performance limitations, and it may be unable to restore the original pattern after training. Therefore, the variational autoencoder adds some noise from a normal distribution sampling into autoencoder training to improve the results, as shown in Figure 4.

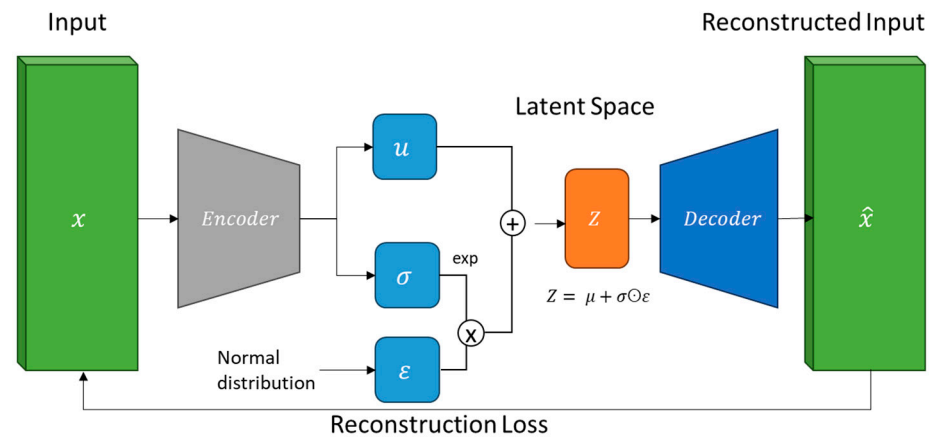


Figure 4. VAE model.

2.6. Bidirectional LSTM (BLSTM)

Our previous paper [3] introduced bidirectional LSTM [18,19] for chipping occurrence prediction. It simultaneously inputs time series into individual LSTMs in forward and backward ways for training, as shown in Figure 5. Inputting the data forward into the LSTM model (denoted Forward LSTM) learns how the generated past data appear in the present data to deduce the causal relationship between each other. Similarly, data can also go backward into the LSTM model (denoted Backward LSTM) to learn the relationship between future data and present data. Finally, we merge the predicted results of the forward and backward outputs by weighted averaging or summing. This way, we can obtain better prediction accuracy than a one-way LSTM model.

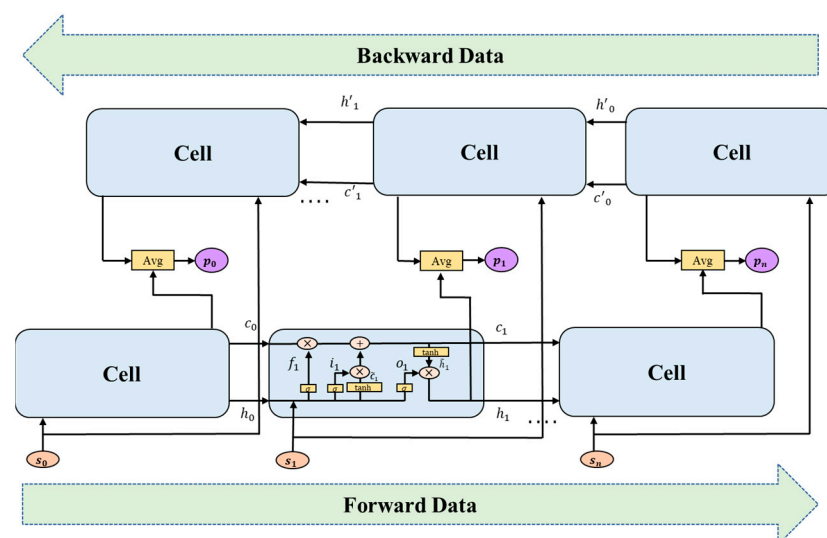


Figure 5. BLSTM architecture.

After dimensionality reduction, we transform every vector of the critical dicing parameters into a single one-dimensional indicator used as an input datum of the forward or

backward sequence in the BLSTM model. In Figure 5, we use the BLSTM model to predict the next indicator value at a time. In Figure 5, we can see that after data enter the model, parameter c_0 is the long-term memory data of the previous dicing parameters, h_0 is the prediction result of the last series time, S_1 is the dicing parameter signal of the current time series, and c_0 will pass A forgetting gate of f_1 , in which c_0 will determine its forgetting proportion with the value calculated through the Sigmoid activation function between h_0 and S_1 . After that, h_0 and S_1 will pass through a memory gate represented by i_1 through the value of tanh of h_0 and S_1 to determine which information to memorize. The data that passed the input gate will be added to c_0 to become c_1 . They will pass through an output gate represented by o_1 , and c_1 will use tanh to decide whether to obtain the output value of the current cell.

2.7. Model-Agnostic Meta-Learning (MAML)

Meta-learning [20], learning to learn, was born with the expectation of human learning ability. Meta-learning is expected to benefit the ability to “learn to learn” and quickly learn new tasks based on attained existing knowledge. Training results are often poor when we train a model due to insufficient data. Meta-learning can learn through the classification experience of previous tasks and quickly adapt to a new task through previously learned data. In machine learning, the training unit is a set of collected data used to optimize the model. The system usually divides the acquired data into a training data set, a verification data set, and a test data set. Meta-learning divides the training unit into two levels. The first-level training unit is the task. Meta-learning requires preparing many tasks for the base learner and the data corresponding to each task to learn the meta learner (generalization model). The second-level training unit is a new task that uses a small amount of new data to achieve rapid convergence and optimize meta learner model parameters.

Instead of a deep learning model, MAML [20] is a framework that provides a meta learner for training base learners. The meta learner here is the essence of MAML and is used to learn, while the base learner is a fundamental mathematical model trained on the target data set and practically used for prediction tasks. Most deep learning models can be embedded in MAML seamlessly as a base learner, and we apply reinforcement learning to MAML. Such a way is called model-agnostic in MAML, as shown in Figure 6.

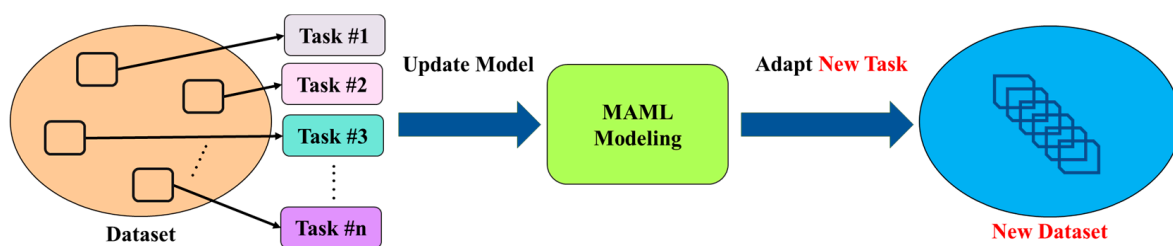


Figure 6. Model-agnostic meta-learning (MAML) execution flow.

2.8. Ensemble Learning

Ensemble learning [21] is a machine learning method that combines multiple different models as base learners to boost the overall prediction accuracy. The main idea of this method is to form a powerful ensemble model by combining various basic models (that is, a learner with a single prediction ability slightly better than random guessing) to improve the overall prediction accuracy and enhance generalization ability. The various frameworks of ensemble learning include bagging, boosting, blending, and stacking. The advantage of ensemble learning is combining the merit of multiple models, reducing the risk of overfitting, improving accuracy, and enhancing overall output stability. It can widely solve machine-learning problems, including regression, classification, feature selection, and anomaly detection. This method is valid for improving prediction accuracy significantly in practical applications and thus has received widespread attention in the industry. The accuracy comparison between ensemble learning uses basic machine learning models and

various CNN models for nine analog and digital modulation signals, where CNN has higher SNR tolerance and better classification than the ensemble learning algorithm [21].

Nevertheless, if we replace basic machine learning models with deep learning models in ensemble learning, ensemble learning performance will outperform the CNN model. In addition, ensemble learning performs well for anomaly detection [22]. This study adopts the blending method in ensemble learning, as shown in Figure 7.

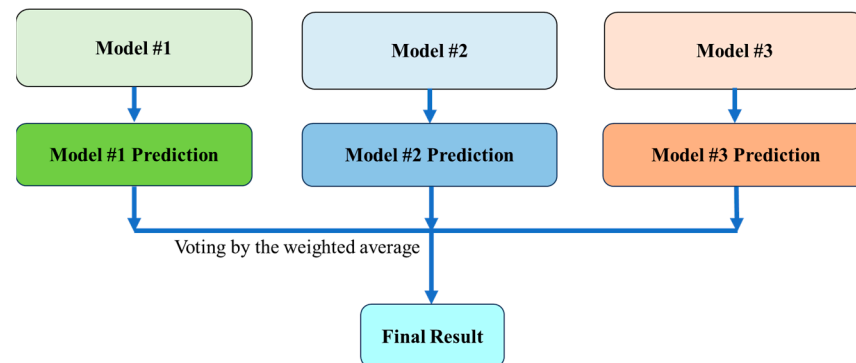


Figure 7. Blending ensemble learning.

3. Method

In the previous work, we implemented importance analysis, random forest, and Barnes–Hut t-SNE dimensionality reduction to obtain critical dicing parameters. The wafer dicing machine retrieved information about 143 wafer dicing parameters corresponding to the chipping position. Then, we applied important analysis to pick up the ten most important dicing parameters related to wafer chipping. Next, the random forest method further filtered out eight critical parameters from the ten important parameters. Finally, we used the eight critical parameters to build the chipping occurrence prediction model. The purpose of selecting parameters from data analysis is to let readers understand the modeling method.

Then, trained bidirectional long short-term memory (BLSTM) can predict wafer chipping occurrence successfully in a single dicing machine. However, each dicing machine of the same type may produce unevenly distributed non-IID dicing signals, which may lead to the undesirable result that a pre-trained model trained by dicing machine #1 cannot effectively predict chipping occurrence in dicing machine #2. This study adopts an ensemble meta-learning-based model that can evaluate many dicing machines with high stability and accuracy for chipping prediction to realize the cross-machine use prediction tool. Here, we will introduce several base learners, such as the hidden Markov model (HMM), the variational autoencoder (VAE), and BLSTM, to form an ensemble learning.

With mainly unsupervised learning and well-performed results in generative AI, the variational autoencoder (VAE) can find subtle abnormal signals by restoring the signal distribution method. If it has detected such abnormal signals, we can infer a high probability of chipping occurrence due to such a model combining the concepts from autoencoder and variational inference to form generative learning. With the stochastic behavior in discovering hidden states and changing processes, the hidden Markov model (HMM) can better find the hidden parameters than the ordinary Markov model to predict the future state accurately. HMM can also perform the applications of generative AI well due to its capability to predict and assess weak nonlinear signals from changes in machinery equipment. Therefore, this study combines HMM, VAE, and BLSTM models. It uses ensemble meta-learning by voting weighted averages between their results to predict large-scale wafer chipping occurrence and achieve the best outcome.

3.1. Exploring Critical Parameters

In our previous work [3], we collected data sheets related to wafer dicing from a semiconductor company in Kaohsiung, Taiwan. The data sheets showed the dicing position coordinates of 112 wafers; the coordinates of each dicing position marked the corresponding kerf, cutline, and channel number. People can look for the parameter codes according to the channel number and cutline indicated in a signal summary table. After that, you can search a data table using the parameter code, then query the data table and retrieve the information about 143 wafer dicing parameters corresponding to the chipping position.

Our previous study performed importance analysis to determine ten important parameters of wafer dicing, as listed in Table 1. Next, this study conducts correlation analysis on these parameters to verify that the spindle current_Z1 and current_Z2 will significantly impact the yield of wafer dicing. Then, the random forest estimates how likely these ten important parameters can affect chipping phenomena. According to ten important dicing parameters applied to the random forest, the estimation accuracy for chipping coverage area less than 30% can achieve 87%. This result shows it is better than using all the important dicing parameters with an estimation accuracy of 78%.

Table 1. Important parameter pick-ups.

No.	Parameter Title	Series No.
1	TDS: Sig:DFD6560_Spindle_SpindleCurrent_Z1 (postRun)	Nil
2	TDS: Sig:DFD6560_Spindle_SpindleCurrent_Z2 (postRun)	Nil
3	TDS: Tool: SV_1555_CuttingWaterStatusZ1_WATERF;	SVID_1555
4	TDS: Tool: SV_1556_CuttingWaterStatusZ2_WATERF2;	SVID_1556
5	TDS: Tool: SV_1772_AnalogFlowSprayNozzleZ1_AVALWATER3 (L);	SVID_1772
6	TDS: Tool: SV_1773_AnalogFlowKerfNozzleZ2_AVALWATER4 (L);	SVID_1773
7	TDS: Tool: SV_1775_AnalogFlowShowerNozzleZ2_AVALWATER6 (L);	SVID_1775
8	TDS: Tool: SV_1752_AnalogPressureMainAir_AVALPRESS0 (MPa);	SVID_1752
9	TDS: Tool: SV_1753_AnalogPressureCleanAir_AVALPRESS1 (MPa);	SVID_1753
10	TDS: Tool: SV_1785_AnalogPressAtomizingNozzleClnAir_AVALPRES9 (MPa);	SVID_1785

The time series analysis used in our previous work [3] can check the data distribution relationship between normal conditions and chipping phenomena. We can match different wafers to examine whether this relationship has regular behavior. For example, we found that backside wall chipping may occur when the wafer's cleaning gas emission parameter SVID_1752 is lower than 586 during the wafer dicing process. In addition, we found that if the air pressure of parameter SVID_1753 changes too much, it can easily cause this backside wall chipping phenomenon as well.

Our previous work [3] inspected the judgment conditions of each important parameter in different decision trees. According to the judgment conditions of each node in the decision tree, we can observe different parameter values that represent normal or chipping situations. According to the parameter values within the judgment conditions, people can better understand which important parameters influence wafer chipping occurrence. We find that parameter SVID_1772 of the node in the decision tree has eight data values greater than 1112.5, and we consider six of them to be chipping phenomena. Therefore, important parameter SVID_1772 has an important influence on determining whether chipping occurs on a wafer. In the random forest estimation, we filter ten important parameters selected from the importance analysis into the eight most important parameters afterward. Eight critical dicing parameters are SpindleCurrent_Z1, SpindleCurrent_Z2, SVID_1772, SVID_1773, SVID_1775, SVID_1752, SVID_1753, and SVID_1785. Furthermore, we also carried out dimensionality reduction using PCA or Barnes–Hut t-SNE for this eight-dimensional parameter vector to a one-dimensional condensed indicator. Then, we realized a heatmap analysis of the indicators to describe the potential chipping occurrence afterward. As a result, Barnes–Hut t-SNE can more significantly reduce dimensionality because its data changes are more sensitive than PCA.

3.2. Dimensionality Reduction

The dicing signals can find critical parameters of wafer dicing in the previous work through importance analysis, correlation analysis, and random forest. It turns out to be a high-dimensional vector, and we make the most significant effort to reduce its dimensionality to a one-dimensional indicator to facilitate the subsequent use of indicators to detect and predict the wafer-chipping phenomenon. Different important dicing parameters with a gap value that is too large may cause the smaller one to be ignored after dimensionality reduction. Thus, the minimax method in Equation (6) standardizes the critical dicing parameters. Equation (6) proportionally adjusts each dicing parameter value within $(0, 1]$, where x_{nom} represents the normalized parameter, x_{max} stands for the maximum parameter, x_{min} indicates the minimum parameter, and x is the current parameter.

$$x_{nom} = \frac{x - x_{min}}{x_{max} - x_{min}}, \quad x_{nom} \in [0, 1] \quad (6)$$

In our previous study, we applied the Barnes–Hut t-SNE dimensionality reduction to the critical dicing parameters because the dimension of the critical dicing parameter vector is not very high. Figure 8 gives the execution flow of the Barnes–Hut t-SNE dimensionality reduction. The time complexity of this computation of the Barnes–Hut t-SNE is $O(n \log^n)$ less than the general t-SNE requiring $O(n^2)$, and it is more practical in this case. An amount of 112 dicing wafers will give 223,990 important characteristic parameters, and these input data denote $x_{h,m}$, where h represents the wafer number and m stands for the parameter number. In Figure 8, Perp determines how many similarities we find in the Barnes–Hut t-SNE dimensionality reduction. The larger the data volume, the higher the Perp is usually set. On the contrary, if the setting is too high in Perp when the amount of data is small, it could cause many dots to be connected too closely, making it impossible to detect subtle changes. p_{ij} represents an approximating probability density function (PDF) of a Gaussian distribution in high-dimensional data when we execute a dimensionality reduction a time. Putting p_{ij} together can construct a high-dimensional PDF matrix $P_{n,n}$, and then this flow will randomly generate an initial low-dimensional output matrix $Y_{n,n}$ using t distribution density function PDF to obtain q_{ij} , where Equation (3) can calculate q_{ij} . We can use q_{ij} to form an output matrix $Y_{n,n}$, and then Equation (4) calculates the loss function C through KL divergence. The closer the C value is to 1, the closer the distance between the two points is; the closer the C value is to 0, the further the distance between the two points is. The gradient descent in Equation (5) updates $Y_{n,n}$.

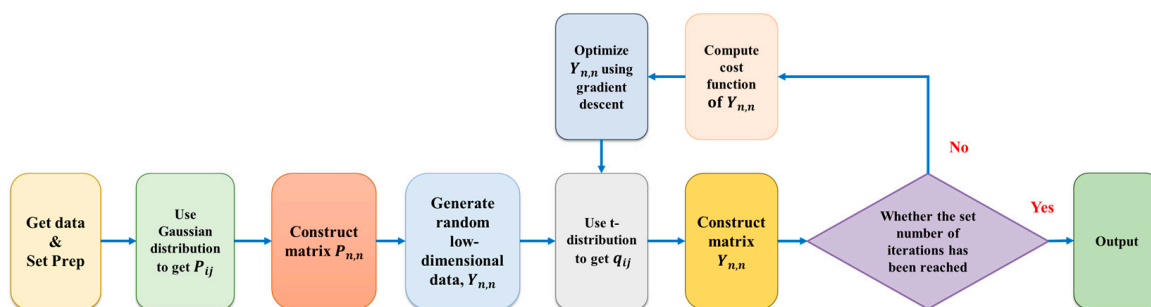


Figure 8. Barnes–Hut t-SNE dimensionality reduction flow.

3.3. Model-Agnostic Meta-Learning (MAML)

The model-agnostic meta-learning (MAML) framework consists of an inner and outer loop. First, the inner loop builds multiple execution threads to run reinforcement learning simultaneously, establishing multiple tasks and policy networks (θ) responding to different environments. Next, the state, action, reward, and loss generated in each environment and the updated policy network parameters (θ') are stored in iteration replay. We choose trust region policy optimization (TRPO) in the outer loop to find the best strategy. The

trajectory reuse strategy of the TRPO algorithm improves the utilization of samples. It ensures that reinforcement learning will not affect the learning effect of the model due to changes in strategy during the training process. This algorithm also delimits the trusted policy learning area to ensure the stability and effectiveness of policy learning, as shown in Figure 9.

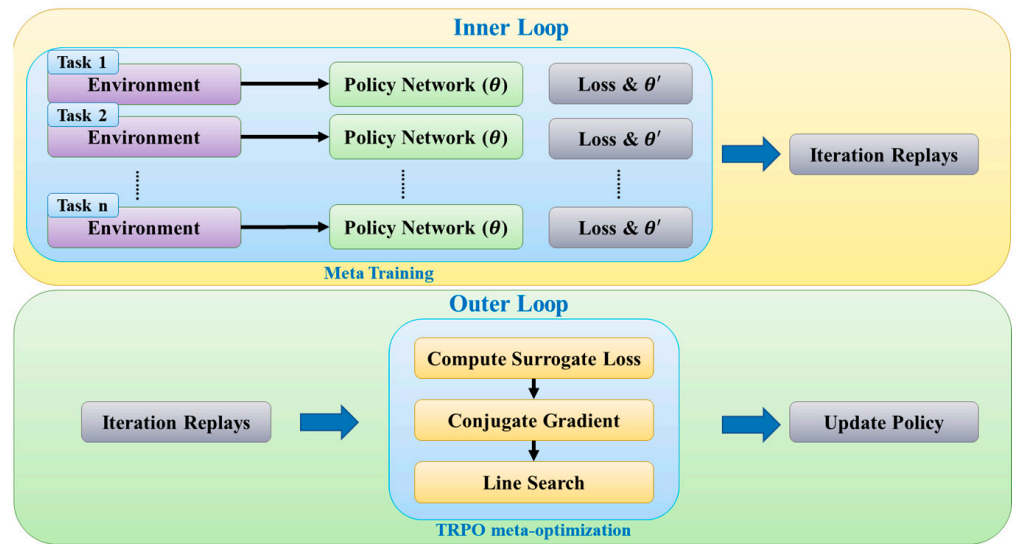


Figure 9. MAML training process.

3.4. Chipping Prediction Model

This study introduces three different MAML models, i.e., HMM, VAE, and BLSTM, as base learners (i.e., prediction model) and incorporates ensemble learning with the MAML framework to establish ensemble meta-learning, as shown in Figure 10. Based on the MAML framework, we train HMM, VAE, and BLSTM in several application domain tasks and validate them by voting on weighted average results. The voting weighted average of all models' outputs can evaluate the ensemble output of chipping prediction, which calculates the chipping occurrence prediction from all the weighted average predictions of each model obtained from the MAML learning. Equation (7) calculates the weight of each prediction model ω_k for ensemble learning, where k represents a specific k th prediction model, te_k stands for the training error of the k th prediction model obtained from the MAML learning, and m is the number of all prediction models. Next, Equation (8) computes the weighted average of the outputs of all prediction models $out_{ensemble}$, where ω_k denotes the weight of each prediction model, out_k indicates the output of the k th prediction model obtained from the MAML learning, k shows a specific the k th prediction model, and m is the number of all prediction models.

$$\omega_k = 1 - \frac{te_k}{\sum_{k=1}^m te_k}, \text{ where } k = 1, 2, \dots, m; \sum_{k=1}^m \omega_k = 1 \tag{7}$$

$$out_{ensemble} = \sum_{k=1}^m \omega_k \cdot out_k \tag{8}$$

In the testing, three trained models will give the inference result out of the voting weight average of the individual output of each meta learner, as shown in Figure 10. The time complexity of this computation of the HMM, VAE, and BLSTM is $O(n \log^n)$, $O(n^2)$, and $O(n^4)$, where n is the number of multiplications of each unit in this case.

Some issues that arise in the training data set are collected data missing, data showing a nonlinear distribution, and correlated hidden parameters lost. One solution is to increase the training data to enhance the information in the training data set. Therefore, our previous work [3] in a wafer chipping coverage area of less than 30% divided the data

set into four groups, including chipping areas of 10%, 10~15%, 15~20%, and 20~30%. Within these groups, we then carry out an average pooling of four sampled data at the same corresponding position in each group to generate an extra data set that can deliver hidden information between data and increase the amount of training data. After the dimensionality reduction, this study imports indicators into ensemble meta-learning to predict the potential large-scale chipping occurrence implemented by HMM, VAE, and BLSTM models, as shown in Figure 10. With a range of indicator inputs, a training model reaches a loss (error) of 0.1116 in the training phase, as shown in Figure 11. Once the trained model passes the testing, this model exploits to predict the chipping occurrence afterward during wafer dicing at the same machine, as shown in Figure 12. In Figure 12, a specific model, such as HMM, VAE, or BLSTM, has been trained entirely at the early stages of the wafer dicing process. Then, it can accurately predict the future indicator that will reveal the occurrence of chipping shortly. In such a way, it can help detect a chipping occurrence or show the trend of potential chipping soon.

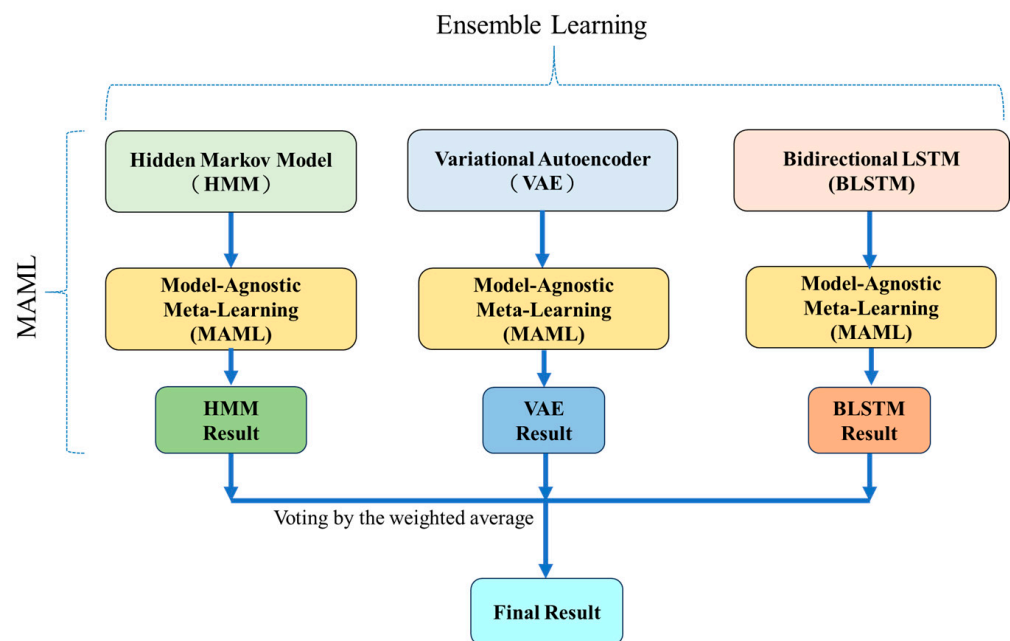


Figure 10. Ensemble meta-learning model.



Figure 11. Loss and accuracy in the training phase of the ensemble meta-learning model.

This study utilizes Anaconda to build the experimental environment and set model hyperparameters, where we set the hidden layer [100, 100], the adapt learning rate 0.5, the number of iterations 100, the meta batch size 10, the number of workers 10, and the cuda 1. Each iteration creates a record file to log the reward and accuracy of the currently trained model and observe the training status through Tensorboard.

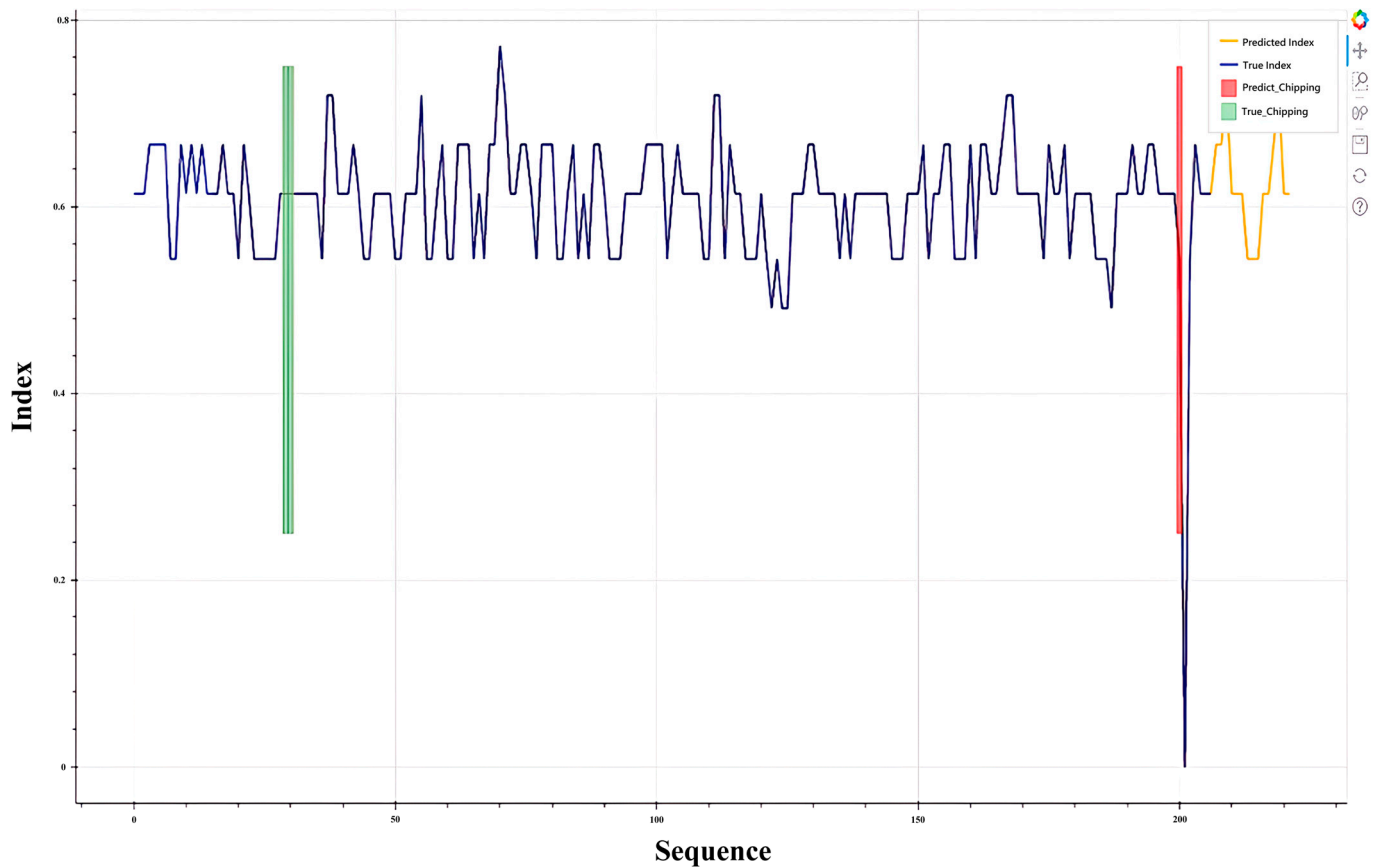


Figure 12. Ensemble meta-learning predicts index changes in advance and discovers chipping phenomena.

3.5. Robust Chipping Prediction System (RCPS)

This study introduces an ensemble meta-learning with the constraint of model-agnostic meta-learning applied to the robust chipping prediction, as shown in Figure 13. First, the network uploads the collected instant dicing signals from different machines (as other tasks) to the in-cloud database. Then, we proceed with data cleaning and aggregation of the dicing signal of each task to form an individual time series so that we can perform data analysis on them. Next, implementing random forest and importance analysis can find the critical dicing parameters. After that, Barnes–Hut t-SNE dimensionality reduction converts every eight-dimensional parameter vector into a single condensed indicator. Finally, we import the processed indicators into the ensemble meta-learning for model training. Then, according to the chipping prediction results, fab decides whether to warn the process personnel to pause the current operation and adjust the critical dicing parameter to reduce the probability of chipping afterward.

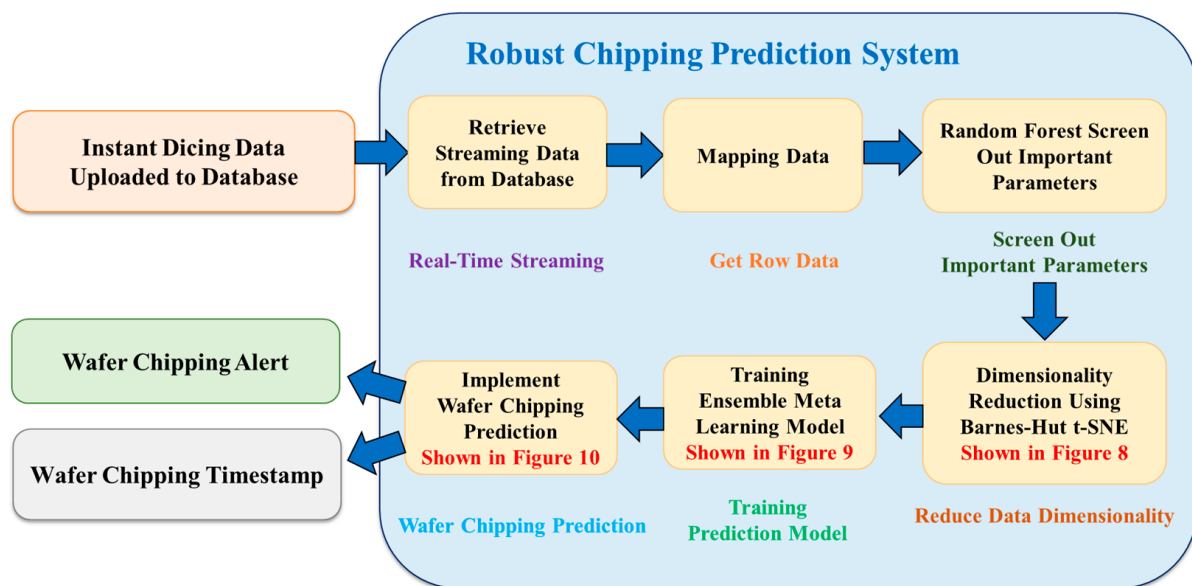


Figure 13. Chipping prediction flow.

4. Experiment Results and Discussion

4.1. Experimental Procedure and Environment

Our previous work [3] adopted PCA and Barnes–Hut t-SNE methods to reduce signal dimensionality and acquire respective condensed indicators. After that, we utilized indicators to train the BLSTM model for the prediction of chipping occurrences implemented. The BLSTM model introduced in our previous paper [3] acts as a baseline compared with this study’s proposed approach in the following experiment. The proposed approach in this paper will put these indicators into the ensemble meta-learning to predict backside wall chipping and check the prediction accuracy of chipping during dicing. Once the predictor detects the possible chipping occurrence, the system should warn the operator to adjust critical dicing parameters promptly to avoid chipping occurrence during the dicing process. An ablation study investigates the performance of different combination ensemble models in machines #1 and #2 individually to understand the component’s contribution to the overall prediction. Finally, comparing chipping results with or without dimensionality reductions demonstrates the significant contribution of this study.

Our previous work [3] showed that the wafer dicing machine was DISCO DFD6560 manufactured by Disco Corporation, Tokyo, Japan with hardware specifications. A table in our previous work also displayed the application software packages. This table shows Pandas 1.1.5 and Numpy 1.19.5 packages for data preprocessing, PCA and t-SNE for dimensionality reduction, Scikit-learn 0.23.2 for machine learning, Tensorflow 2.6.2 and Keras 2.6.0 for deep learning, and Matplotlib 3.3.4 and Pyplot 5.5.0 for output visualization.

4.2. Data Collection and Dimensionality Reduction

According to the importance analysis in the previous chapter, there are seven critical parameters for wafer dicing, including a spindle current, three types of jet water flow, and three types of clean gas emission. Here, we put them as an eight-dimensional vector and want to convert this vector into a one-dimensional condensed indicator by dimensionality reduction. According to the settings for parameter dimensionality reduction in our previous work [3], we used PCA and Barnes–Hut t-SNE methods to reduce an eight-dimensional vector to a one-dimensional condensed indicator. The dimensionality reduction obtained 231,990 data concerning the critical parameters from 112 pieces of wafer dicing, where 176,900 data were obtained from machine #1 from 85 pieces of wafer dicing, and 55,090 data were attained from machine #2 from 27 pieces of wafer dicing.

4.3. Modeling Using Ensemble Meta-Learning and Chipping Prediction

For data allocation in machine #1, we arranged a training data set of 167,993 signals and a testing data set of 8907 signals, which collected the frontend dicing signals and their corresponding backend dicing signals from 85 different wafers. Next, this study set the model and training parameters for HMM, VAE, and BLSTM base learners in the following experiments. Concerning the HMM setting in the training phase of machine #1, we set the training parameters to include two states in the model, 150 iterations performed during training, and a full covariance type. Regarding the VAE setting in the hyperparameter, the encoder consists of four layers. The first layer is a convolutional layer with 32 pieces of 3×3 kernels, and the second is a convolutional layer with 64 pieces of 3×3 kernels. Then, the third layer is a dense layer with 32 neurons, and the fourth layer is a dense layer with 32 neurons. The decoder is an encoder image with a symmetrical four layer. All activation functions are ReLU in the VAE model. In the training phase of machine #1, this experiment set the training parameter epochs to 100, batch_size 128, and loss_funtion "Kullback-Leibler divergence". As for the BLSTM setting in the training phase of machine #1, every single LSTM model constructs three layers and a total of 640 neurons, as shown in Figure 5. In Figure 5, the optimizer is adaptive moment estimation (Adam), the loss function defines mean-square error (MSE), the activation function *tanh* means the hyperbolic tangent, and the activation function σ indicates the sigmoid function. If the accuracy does not increase significantly in ten consequent training rounds, the training will terminate due to an early stopping setting. To return short-term output results between multiple units, we set return_sequences to true. This experiment sets the epoch to 50 and the batch size to 128.

For wafer dicing in machine #1, we view the collected data sets concerning chipping coverage area 10%, 10~15%, 15~20%, 20~30%, and its generated data set as tasks #1, #2, #3, #4, and #5, respectively, applied to MAML modeling. The prediction models using Barnes–Hut t-SNE dimensionality reduction give the prediction accuracy, as shown in Table 2. In Table 2, ensemble meta-learning achieves the best accuracy in both categories. The hidden Markov model has the worst prediction accuracy in Class I, while the variational autoencoder has the worst in Class II. According to the ablation study for ensemble meta-learning (EML), this experiment uses different combinations of HMM, VAE, and BLSTM models to verify the validity of the approaches. In the experiment, the ensemble meta-learning using HMM and BLSTM denotes EML_HMM+ BLSTM, ensemble meta-learning using VAE and BLSTM abbreviates EML_VAE+BLSTM, and ensemble meta-learning using HMM, VAE, and BLSTM marks EML_HMM+VAE+BLSTM.

Table 2. Comparison of accuracy of various meta learners in machine #1.

Chipping Area in Machine #1	Models						
		HMM	VAE	BLSTM *	EML_ HMM + BLSTM	EML_ VAE + BLSTM	EML_ HMM + VAE + BLSTM
Class I: less than 30%		0.7622	0.8623	0.9234	0.9267	0.9275	0.9323
Class II: more than 30%		0.6213	0.5431	0.8216	0.8222	0.8227	0.8233

* is a baseline and the result of our previous work [3].

For wafer dicing in machine #2, we treat the collected 55,090 data as new data sets, where we arrange a training data set of 52,336 signals and a testing data set of 2754 signals. This use case finds them to have equal chipping numbers distributed between 10~20% and 20~30% of the chipping coverage area, and its newly generated data set as new tasks #1, #2, and #3, respectively. According to a trained prediction model (called meta learner) obtained in machine #1, modeling machine #2 uses the collected data to fine-tune and test meta learner to find an optimal prediction model for machine #2. This use case is not a traditional transfer learning to use the data set of machine #2 to fine-tune and test a pre-trained model obtained in machine #1. Similarly, such a MAML approach can also apply to modeling the same type of other dicing machines #3, #4, #5, and #n. According to

a trained prediction model (called meta learner) obtained in machine #1, modeling in every machine can effectively use its corresponding collected data to fine-tune and test meta learner to find an optimal prediction model for every machine. The prediction models using Barnes–Hut t-SNE dimensionality reduction deliver the prediction accuracy, as shown in Table 3. We find that the prediction accuracy of each model decreases significantly during wafer dicing in machine #2. Nevertheless, the predictive effect of ensemble meta-learning is less affected and can still maintain higher accuracy. This mechanism confirms that the proposed approach can preserve the highly predictive accuracy, achieving a robust prediction behavior.

Table 3. Comparison of accuracy of various meta learners in machine #2.

Chipping Area in Machine #1	Models						
	HMM	VAE	BLSTM *	EML_ HMM + BLSTM	EML_ VAE + BLSTM	EML_ HMM + VAE + BLSTM	
Class I: less than 30%	0.6412	0.7123	0.8579	0.8727	0.8901	0.9027	
Class II: more than 30%	0.5130	0.5122	0.7813	0.7980	0.8095	0.8154	

* is a baseline and the result of our previous work [3].

After completing the dimensionality reduction, either PCA or Barnes–Hut t-SNE to obtain condensed indicators, we train different condensed indicators with an ensemble meta-learning under the constraint of model-agnostic meta-learning, as shown in Figure 14. We then import the test data set about indicators into the trained ensemble meta-learning to predict possible chipping during wafer dicing. We compare the prediction accuracy among the different dimensionality reductions, namely without reduction, PCS, and Barnes–Hut t-SNE methods, as listed in Table 4.

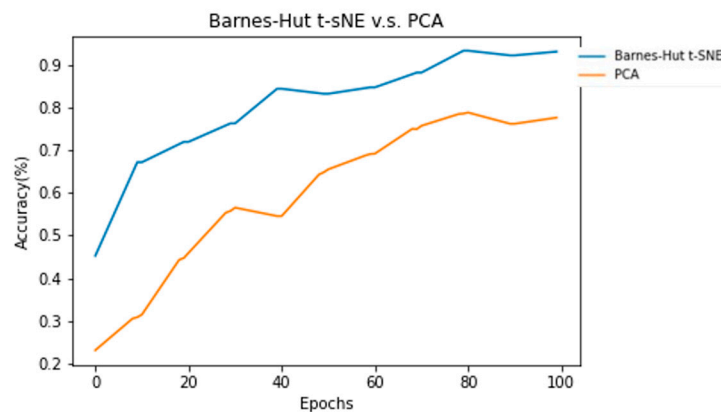


Figure 14. Prediction accuracy of different dimensionality reductions.

Table 4. Accuracy comparison with different dimensionality reductions using ensemble meta-learning in machines #1 and #2, respectively.

Chipping Area of a Wafer	Dimensionality Reduction		
	Without Reduction	PCA	Barnes–Hut t-SNE
Machine #1	Less than 30%	0.8473	0.9323
	More than 30%	0.7424	0.8233
Machine #2	Less than 30%	0.7821	0.9027
	More than 30%	0.6785	0.8154

Table 4 shows the chipping samples from the wafer dicing machine divided into two categories. The first category is the chipping coverage area of less than 30% on the surface of

a single wafer, and the second category is greater than 30%. The ratio of the former samples to the latter is 11:1. By collecting the above chipping samples, we can train the ensemble meta-learning and then obtain a trained model. In the testing, the chipping prediction accuracy of the first category is 10.9% higher than the second category. On the other hand, the chipping data shows a nonlinear distribution, and thus, the chipping prediction accuracy of using the Barnes–Hut t-SNE in the test will be 14.48% higher than PCA.

4.4. Wafer Dicing Results

During wafer dicing, Barnes–Hut t-SNE reduces the dimensionality of the input vector to the condensed information that feeds into the ensemble meta-learning learning to predict possible chipping. Since the kerf is detected to be worn out, it will cause more chipping if not replaced. This study compares two situations between the default setting without adjusting the critical dicing parameters during the dicing process and the tuning setting with timely changing the critical dicing parameters according to our proposed approach. The tuning setting can rapidly adjust critical dicing parameters and check whether it can effectively control the occurrence of large-area wafer chipping. The case in the default setting might lead to the rapid wear of the kerf where the probability of wafer chipping will gradually increase, resulting in changing the kerf after three pieces of wafer dicing [3]. In contrast, in our previous paper [3], the case in tuning setting with single BLSTM model can change the critical dicing parameters appropriately to reduce the probability of chipping occurrence and maintain most diminutive kerf wearing, resulting in eight pieces of wafers in dicing for every kerf using the single BLSTM model referred to. With our proposed ensemble meta-learning approach in tuning critical parameters optimally while wafer dicing, this study achieves better results, achieving ten pieces of wafers in dicing for every kerf, as shown in Figure 15. In Table 5, the proposed method significantly improves the yield of wafer dicing, thereby reducing the manufacturing costs.

Table 5. Comparison of chipping after wafer dicing.

Setting Attribute	Parameters	Default	BLSTM	Ensemble Meta-Learning HMM + VAE + BLSTM
Number of wafers diced before a kerf change is needed		3	8	10
Distribution of backside wall chipping		Whole wafer	Bottom half of a wafer	Right half of a wafer

4.5. Discussion

Random forest can infer the wafer chipping occurrence in dicing, effectively exploring important dicing parameters. In the experiments, this study finds that, for the coverage area in chipping of less than 30% on the surface of a wafer, the prediction accuracy can be as high as 87%. However, random forest can only estimate most defects after the chipping occurs. Still, they cannot do this during the chipping process due to the poor sensitivity of wafer chipping detection and prediction in dicing. The sensitivity of wafer chipping detection and prediction in dicing is poor. Instead, if we test random forest with a wafer covering a chipping area of more than 50%, the estimation accuracy drops to 52%. Such a case can successfully predict fewer chipping during the estimation process. Most predictions are misclassified to normal situations when chipping has already occurred. Suppose you observe node judgment conditions in random forest, in that case, you probably focus on a few key nodes to explain the chipping phenomenon and pick up the critical dicing parameters if needed.

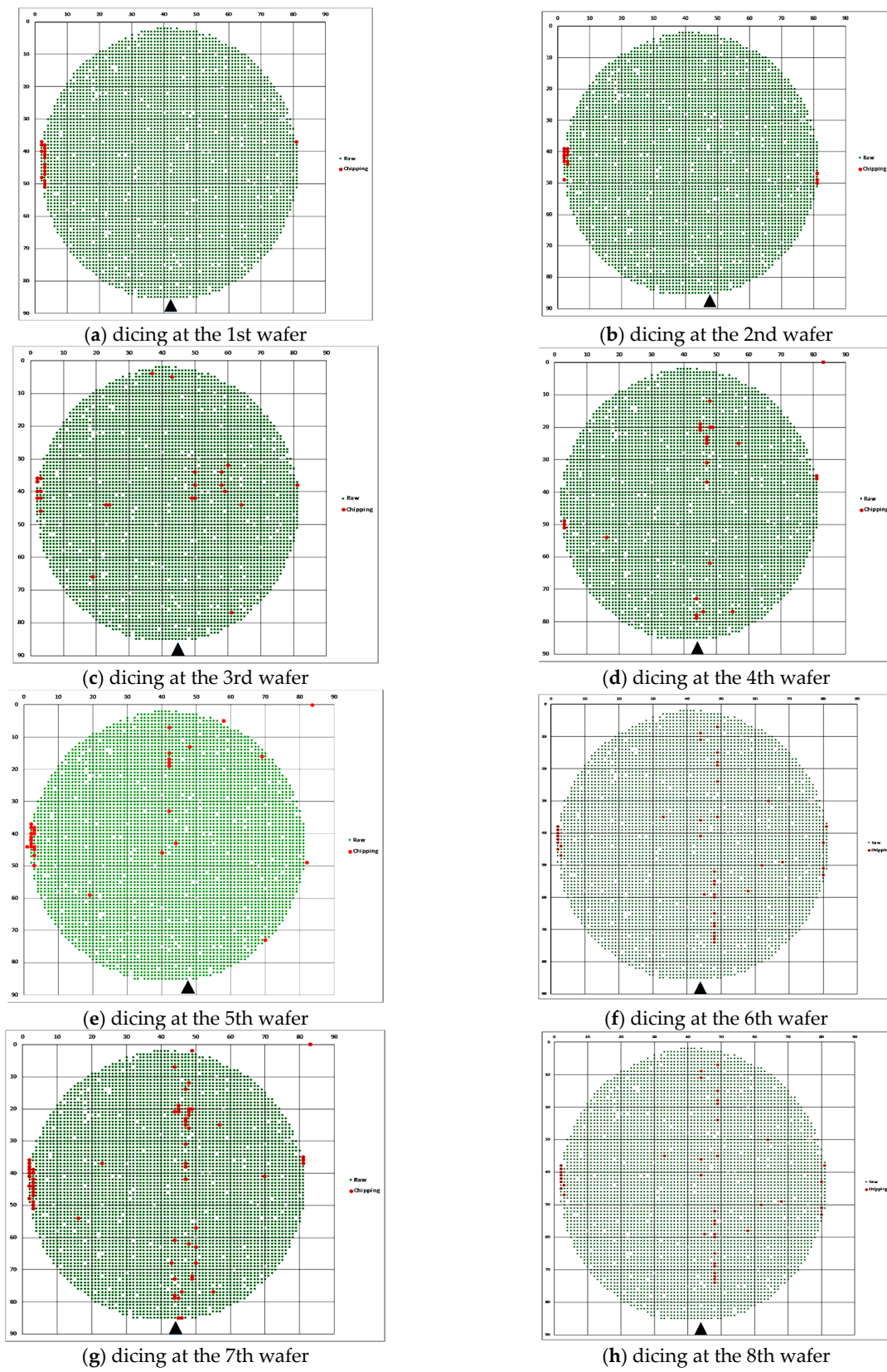


Figure 15. Cont.

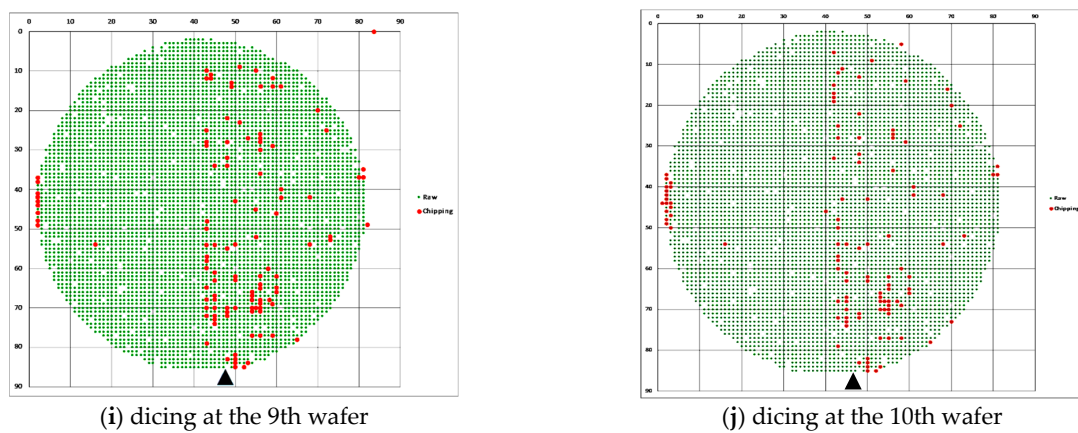


Figure 15. Chipping distributed (red dot indicated) in wafer dicing with parameters.

This study only selects three base learners, such as HMM, VAE, and BLSTM, to construct an ensemble meta-learning in the experiments. Although other applicable base learners have been found, such as ARIMA, sparse transformer, and XLNet, their implementation effects are not sound in the chipping prediction during wafer dicing. Thus, we have not included them in this study's list of base learners. In the future, we hope to find more base learners and train them through model-agnostic meta-learning to obtain better execution results. Even though BLSTM cannot perform very well in chipping prediction in some iterations (or at some time instants), HMM or VAE could achieve better prediction accuracy than BLSTM. Therefore, the overall result can achieve the best prediction accuracy due to the voting by the weighted average of every output among the three models. This can explain why the ensemble meta-learning frame can outperform the individual model output in chipping prediction.

There are two limitations to the experiments. The old type wafer dicing machines cannot give a specific log concerning the outlier or chipping in detail. Therefore, the machine needs manual labeling to mark the chipping coverage area and read each piece of chipping data individually after wafer dicing. It costs much of the workforce to deal with data cleaning and aggregation. In addition, the current approach cannot handle extreme cases successfully, especially the coverage area with more than 70% chipping on the wafer surface. On the other hand, this situation forces the operator to replace the worn kerf compulsorily during wafer dicing.

5. Conclusions

A semiconductor factory needs many machines to achieve the predetermined yield rate during mass production. In our previous study, only a single pre-trained model out of a specific machine to predict chipping in another machine was not practical due to different chipping coverage area distribution in other dicing machines. The main contribution of this study in solving the problem mentioned above is to introduce ensemble meta-learning-based robust chipping prediction that can effectively apply to many dicing machines for chipping prediction with high stability and accuracy. This study proposes ensemble learning incorporating model-agnostic meta-learning to establish an ensemble meta-learning that will vote the weighted average from several meta learners to obtain the high accuracy of chipping prediction with stability across several dicing machines. The goal is to adjust key dicing parameters rapidly and avoid the occurrence of large-scale chipping during wafer dicing. Accordingly, the proposed approach will promote the yield rate of wafer dicing and cut wafer manufacturing costs.

Regarding the prospects of this study, early wafer dicing machines cannot automatically log in data to mark the chipping coverage area when chipping occurs. After the machine dices the wafer, it is necessary to manually keep the chipping coverage areas to find each piece of data on the chipping phenomenon, wasting much of the workforce

manipulating the data. In the future, we look forward to adopting more advanced dicing machines that can perform visual algorithms and related software to automatically mark outliers that cause chipping when dicing wafers on the machine. The machine collects data efficiently and automatically, saving time and effort.

Author Contributions: B.R.C. and H.-Y.M. conceived and designed the experiments; H.-F.T. collected the data set and proofread the manuscript; and B.R.C. wrote the paper. All authors have read and agreed to the published version of the manuscript.

Funding: The National Science and Technology Council fully supported this work in Taiwan, Republic of China, under grant numbers NSTC 112-2622-E-390-001 and NSTC 112-2221-E-390-017.

Data Availability Statement: The Sample Programs for Sample Program.zip data used to support the findings of this study are available at https://drive.google.com/drive/folders/1HibUtORzDRI7taHIC0JOfiPAiobc3q95?usp=drive_link (accessed on 23 February 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Rosa-Zurera, M.; Jarabo-Amores, P.; Lopez-Ferreras, F.; Sanz-Gonzalez, J.L. Comparative analysis of importance sampling techniques to estimate error functions for training neural networks. In Proceedings of the IEEE/SP 13th Workshop on Statistical Signal Processing, Bordeaux, France, 17–20 July 2005; pp. 121–126.
- Onda, H. Framework for wafer level control APC model. In Proceedings of the 2011 e-Manufacturing & Design Collaboration Symposium & International Symposium on Semiconductor Manufacturing (eMDC & ISSM), Hsinchu, Taiwan, 5–6 September 2011; pp. 1–10.
- Chang, B.R.; Tsai, H.-F.; Mo, H.-Y. Detection and Prediction of Chipping in Wafer Grinding Based on Dicing Signal. *Mathematics* **2022**, *10*, 4631. [[CrossRef](#)]
- Khokhar, M.S.; Cheng, K.; Ayoub, M.; Eric, L.K. Multi-Dimension Projection for Non-Linear Data Via Spearman Correlation Analysis (MD-SCA). In Proceedings of the 2019 8th International Conference on Information and Communication Technologies (ICICT), Karachi, Pakistan, 16–17 November 2019; pp. 14–18.
- Dong, Y.-Q. Value Ranges of Spearman's Rho and Kendall's Tau of a Class of Copulas. In Proceedings of the 2010 International Conference on Computational and Information Sciences, Chengdu, China, 17–19 December 2010; pp. 182–185.
- Zhang, Z.; Yang, X. Constructing Copulas on the Parabolic Boundary of Kendall's Tau-Spearman's Rho Region. In Proceedings of the 2010 First ACIS International Symposium on Cryptography and Network Security, Data Mining and Knowledge Discovery, E-Commerce and Its Applications, and Embedded Systems, Qinhuangdao, China, 23–24 October 2010; pp. 324–327.
- Sangwan, A.; Zhu, W.; Ahmad, M.O. Design and Performance Analysis of Bayesian, Neyman–Pearson, and Competitive Neyman–Pearson Voice Activity Detectors. *IEEE Trans. Signal Process.* **2007**, *55*, 4341–4353. [[CrossRef](#)]
- Zhang, Q.T.; Song, S.H. Model Selection and Estimation for Lognormal Sums in Pearson's Framework. In Proceedings of the 2006 IEEE 63rd Vehicular Technology Conference, Melbourne, VIC, Australia, 7–10 May 2006; pp. 2823–2827.
- Jiao, Y.; Vert, J. The Kendall and Mallows Kernels for Permutations. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 1755–1769. [[CrossRef](#)] [[PubMed](#)]
- Xue, D.; Zhong, C.; Zhang, E.; Jiang, W.; Zhang, C. Die Chipping FDC Development at Wafer Saw Process. In Proceedings of the 2021 22nd International Conference on Electronic Packaging Technology (ICEPT), Xiamen, China, 14–17 September 2021; pp. 1–2.
- Kang, S.; Cho, S.; An, D.; Rim, J. Using Wafer Map Features to Better Predict Die-Level Failures in Final Test. *IEEE Trans. Semicond. Manuf.* **2015**, *28*, 431–437. [[CrossRef](#)]
- Schelthoff, K.; Jacobi, C.; Schlosser, E.; Plohmann, D.; Janus, M.; Furmans, K. Feature Selection for Waiting Time Predictions in Semiconductor Wafer Fabs. *IEEE Trans. Semicond. Manuf.* **2022**, *35*, 546–555. [[CrossRef](#)]
- Yamaki, S.; Seki, S.; Sugita, N.; Yoshizawa, M. Performance Evaluation of Cross Correlation Functions Based on Correlation Filters. In Proceedings of the 2021 20th International Symposium on Communications and Information Technologies (ISCIT), Tottori, Japan, 19–22 October 2021; pp. 145–149.
- Meyer, B.H.; Pozo, A.T.R.; Zola, W.M.N. Improving Barnes-Hut t-SNE Scalability in GPU with Efficient Memory Access Strategies. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.
- Zeng, Y.; Lou, Z. The New PCA for Dynamic and Non-Gaussian Processes. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 6–8 November 2020; pp. 935–938.
- Xia, Z.; Chen, Y.; Xu, C. Multiview PCA: A Methodology of Feature Extraction and Dimension Reduction for High-Order Data. *IEEE Trans. Cybern.* **2022**, *52*, 11068–11080. [[CrossRef](#)] [[PubMed](#)]
- Liu, D.; Guo, T.; Chen, M. Fault Detection Based on Modified t-SNE. In Proceedings of the 2019 CAA Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS), Xiamen, China, 5–7 July 2019; pp. 269–273.
- Chatzimpampas, A.; Martins, R.M.; Kerren, A. t-viSNE: Interactive Assessment and Interpretation of t-SNE Projections. *IEEE Trans. Vis. Comput. Graph.* **2020**, *26*, 2696–2714. [[CrossRef](#)] [[PubMed](#)]

19. Aparna, R.; Chitralekha, C.K.; Chaudhari, S. Comparative Study of CNN, VGG16 with LSTM and VGG16 with Bidirectional LSTM Using Kitchen Activity Dataset. In Proceedings of the 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 11–13 November 2021; pp. 836–843.
20. Shao, Y.; Wu, W.; You, X.; Gao, C.; Sang, N. Improving the Generalization of MAML in Few-Shot Classification via Bi-Level Constraint. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 3284–3295. [[CrossRef](#)]
21. Yang, Y.; Zhang, M.; Pei, H. A Comparative Study of Signal Recognition Based on Ensemble Learning and Deep Learning. In Proceedings of the 2023 International Seminar on Computer Science and Engineering Technology (SCSET), New York, NY, USA, 29–30 April 2023; pp. 340–343.
22. Dhibi, K.; Mansouri, M.; Bouzrara, K.; Nounou, H.; Nounou, M. Reduced KPCA based Ensemble Learning Approach for Fault Diagnosis of Grid-Connected PV Systems. In Proceedings of the 2022 19th International Multi-Conference on Systems, Signals & Devices (SSD), Sétif, Algeria, 6–10 May 2022; pp. 841–845.
23. Gao, F.; Huang, T.; Wang, J.; Sun, J.; Hussain, A.; Zhou, H. A Novel Multi-Input Bidirectional LSTM and HMM Based Approach for Target Recognition from Multi-Domain Radar Range Profiles. *Electronics* **2019**, *84*, 535. [[CrossRef](#)]
24. Gong, A.; Chen, C.; Peng, M. Human Interaction Recognition Based on Deep Learning and HMM. *IEEE Access* **2019**, *7*, 161123–161130. [[CrossRef](#)]
25. Zhao, Y.; Zhang, X.; Shang, Z.; Cao, Z. DA-LSTM-VAE: Dual-Stage Attention-Based LSTM-VAE for KPI Anomaly Detection. *Entropy* **2022**, *24*, 1613. [[CrossRef](#)] [[PubMed](#)]
26. Kim, T.; Lee, D.; Hwangbo, S. A Deep-Learning Framework for Forecasting Renewable Demands Using Variational Auto-Encoder and Bidirectional Long Short-Term Memory. *Sustain. Energy Grids Netw.* **2024**, *38*, 101245. [[CrossRef](#)]
27. Thiry, L.; Zhao, H.; Hassenforder, M. Categorical Models for BigData. In Proceedings of the 2018 IEEE International Congress on Big Data (BigData Congress), San Francisco, CA, USA, 2–7 July 2018; pp. 272–275.
28. Garibo-Morante, A.A.; Tellez, F.O. Univariate and Multivariate Time Series Modeling using a Harmonic Decomposition Methodology. *IEEE Lat. Am. Trans.* **2022**, *20*, 372–378. [[CrossRef](#)]
29. Li, K.S.-M.; Jiang, X.-H.; Chen, L.L.-Y.; Wang, S.-Y.; Huang, A.Y.-A.; Chen, J.E.; Liang, H.S.; Hsu, C.-L. Wafer Defect Pattern Labeling and Recognition Using Semi-Supervised Learning. *IEEE Trans. Semicond. Manuf.* **2022**, *35*, 291–299. [[CrossRef](#)]
30. Tsuda, T.; Inoue, S.; Kayahara, A.; Imai, S.-i.; Tanaka, T.; Sato, N.; Yasuda, S. Advanced Semiconductor Manufacturing Using Big Data. *IEEE Trans. Semicond. Manuf.* **2015**, *28*, 229–235. [[CrossRef](#)]
31. Fan, S.-K.S.; Hsu, C.-Y.; Tsai, D.-M.; He, F.; Cheng, C.-C. Data-Driven Approach for Fault Detection and Diagnostic in Semiconductor Manufacturing. *IEEE Trans. Autom. Sci. Eng.* **2020**, *17*, 1925–1936. [[CrossRef](#)]
32. Sunny, M.A.I.; Maswood, M.M.S.; Alharbi, A.G. Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model. In Proceedings of the 2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES), Giza, Egypt, 24–26 October 2020; pp. 87–92.
33. Yang, S. Research on Network Behavior Anomaly Analysis Based on Bidirectional LSTM. In Proceedings of the 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 15–17 March 2019; pp. 798–802.
34. Liu, D.; Wang, J.; Shang, S.; Han, P. MSDR: Multi-Step Dependency Relation Networks for Spatial-Temporal Forecasting. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD). Convention Center, Washington, DC, USA, 14–18 August 2022; pp. 1042–1050.
35. Ou, J.-J.; Sun, J.-H.; Zhu, Y.-H.; Jin, H.-M.; Liu, Y.J.; Zhang, F.; Huang, J.-Q.; Wang, X.B. STP-TrellisNets: Spatial-Temporal Parallel TrellisNets for Metro Station Passenger Flow Prediction. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM), Online, 19–23 October 2020; pp. 1185–1194.
36. Deng, S.; Rangwala, H.; Ning, Y. Robust Event Forecasting with Spatiotemporal Confounder Learning. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD). Convention Center, Washington, DC, USA, 14–18 August 2022; pp. 294–304.
37. Yumeng, C.; Yinglan, F. Research on PCA Data Dimension Reduction Algorithm Based on Entropy Weight Method. In Proceedings of the 2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), Taiyuan, China, 23–25 October 2020; pp. 392–396.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.