Evaluation of machine learning classifiers in faulty die prediction to maximize cost scrapping avoidance and assembly test capacity savings in semiconductor integrated circuit (IC) manufacturing

Cite as: AIP Conference Proceedings **2138**, 040010 (2019); https://doi.org/10.1063/1.5121089
Published Online: 21 August 2019

Azlan Faizal Mohd Fazil, Izwan Nizal Mohd Shaharanee, and Jastini Mohd Jamil

Evaluation of Machine Learning Classifiers in Faulty Die Prediction to Maximize Cost Scrapping Avoidance and Assembly Test Capacity Savings in Semiconductor Integrated Circuit (IC) Manufacturing

Azlan Faizal Mohd Fazil [a], Izwan Nizal Mohd Shaharanee [b] and Jastini Mohd Jamil [c]

School of Quantitative Sciences, Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia

[a] Corresponding author: azlanfaizal77@gmail.com
[b] nizal@uum.edu.my
[c] jastini@uum.edu.my

Abstract. Semiconductor manufacturing is a complex and expensive process. The semiconductor packaging trending towards for more complex package with higher performance and lower power consumption. The silicon die is manufactured using smaller fab process technology node and packaging technology is using more complex and expensive packaging. The semiconductor packaging trend has evolved from single die packaging to multi die packaging. The multi die packaging requires more processing steps and tools in assembly process as well. All these factors cause cost per unit to increase. With this multi die packaging, it results higher loss in production yield compared to single die packaging because overall yield now is a function of multiplication of yield for each individual die. If any die from the final package tested at Class and found to be faulty not meeting the product specification, even the rest of die still passing the tests, the whole package will still be scrapped. This resulting in wasted good raw material (good die and good substrate) and manufacturing capacity used to assemble and test affected bad package. In this research work, a new framework is proposed for model training and evaluation for the machine learning application in semiconductor test with objective to screen bad die using machine learning before die attachment to package. The model training flow will have 2 classifier groupings which are control group and auto machine learning (ML) where feature selection with redundancy elimination method to be applied on input data to reduce the number of variables to minimum prior modeling flow. The control group will serve as reference. The other group, will use auto machine learning (ML) to run multiple classifiers automatically and only top 3 to be selected for next step. The performance metric used is recall rate at specified precision from ROI breakeven point. The threshold probability that correspond to fixed precision will be set as the classifier threshold during model evaluation on unseen datasets. The model evaluation flow will use 3 different non-overlapped datasets and comparison of classifiers will be based on recall rate and precision rate. This new framework will be able to provide range of possible recall rate from minimum to maximum, to identify which classifier algorithm performs the best for given dataset. The selected model can be implemented into actual manufacturing flow to screen predicted bad die for maximum cost scrapping avoidance and capacity savings.

INTRODUCTION

The semiconductor packaging trending towards for complex package with higher performance and lower power consumption. The silicon die is manufactured using smaller technology node and packaging is using more complex and expensive packaging where multiple die being attached on same substrate. The visualization of semiconductor packaging trend in [7] clearly shows how semiconductor packaging evolved from single die packaging since 1970s to multi die packaging after 2000s. The multi die packaging includes System in Package (SiP), 2D Package, 2.5D Package, and 3D Package [6]. The more complex packaging requires more processing steps and tools in assembly

The 4th Innovation and Analytics Conference & Exhibition (IACE 2019)
AIP Conf. Proc. 2138, 040010-1–040010-6; https://doi.org/10.1063/1.5121089
Published by AIP Publishing. 978-0-7354-1881-3/$30.00

process as well. All these factors cause cost per unit to increase. With this multi die packaging, it results higher loss in production yield compared to single die packaging because overall yield now is a function of multiplication of yield for each individual die. If any die in final package found to be faulty at Class test, even the rest of other die still passing the tests, the whole package will still be rejected and scrapped. This resulting in wasted good raw material (good die and good substrate) and manufacturing capacity used to assemble and test affected bad package.

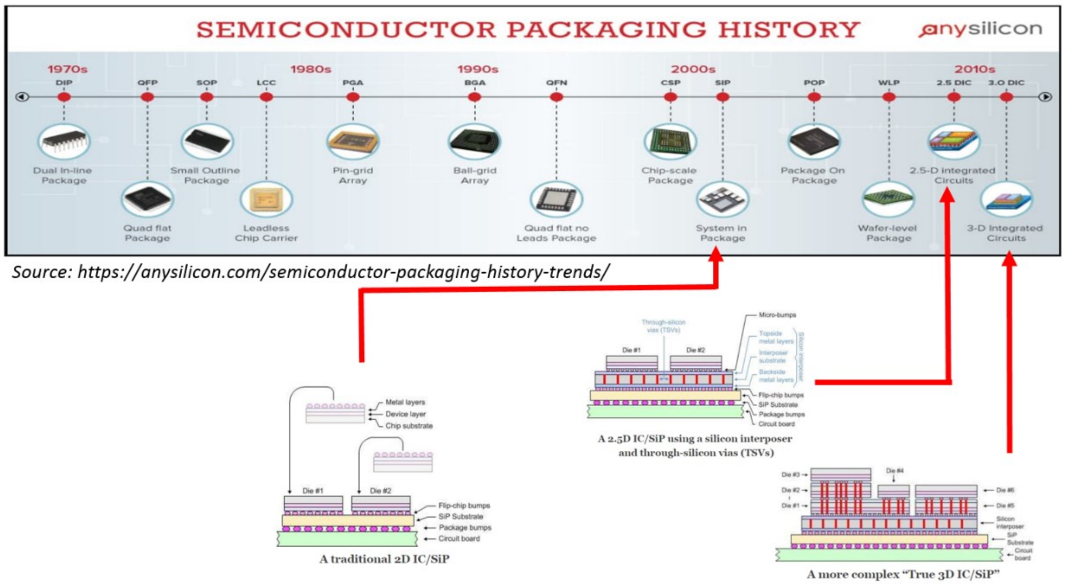


FIGURE 1. Semiconductor Packaging Technology Trend

This research will focus on evaluation of different types of classifiers for unit level prediction using specific semiconductor manufacturing data (Sort and Class) for an existing product that is specially selected to represent typical semiconductor manufacturing test data. The result will provide a reference how does classification algorithms perform compared to each other and gives clear range of opportunity from minimum to maximum. Besides that, another aspect of focus will be on the framework design for model training and evaluation of this application

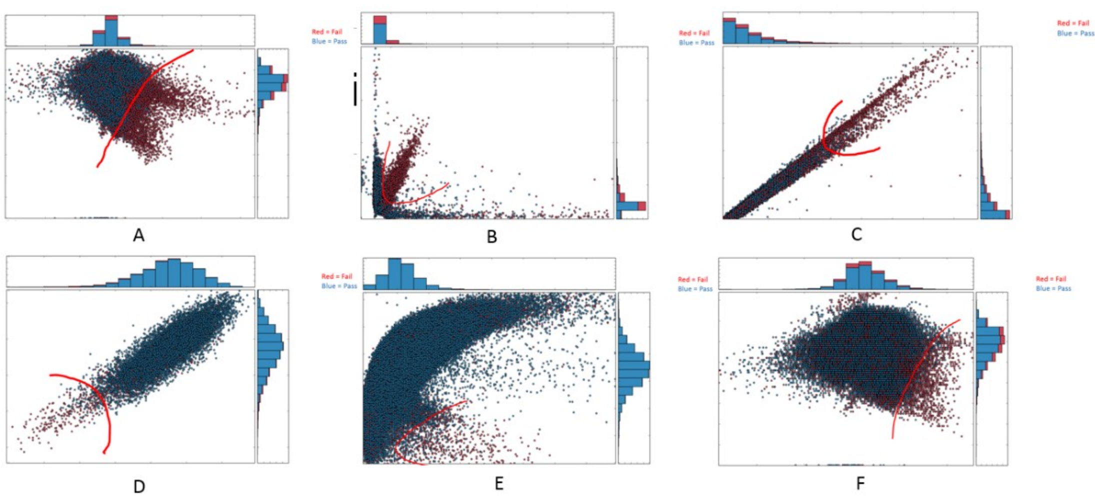


FIGURE 2. Interesting pattern for Pass and Fail die between 2 Sort parameters for multiple products

LITERATURE REVIEW

Machine learning application is not new, there have been many researches and real applications in real world problems in different areas. For our case, it is clear the machine learning application sought is **supervised learning, and classification problem**. As for prediction model building, historical Sort data will be the input and Class data will be the response/output. Since semiconductor testing usually assigns "Pass" or "Fail" result thus becoming a classification problem. The target is to be create predictive model that able to classify incoming die whether it will pass or fail the class test. The chosen model must meet the requirement of having **precision > minimum threshold** so that only bad die will be flagged for scrap prior assembly process and overall process will give positive Return on Investment (ROI). There are hundreds of algorithms available for classification problem. For each type of algorithm, there will be a number of model hyper parameter setups which can be tuned to give optimal result. This indirectly expands the potential list of algorithms from hundreds to thousands. The main focus question in this work is for semiconductor testing dataset, for this unit/die level prediction using Sort and Class data, which algorithm will perform the best and what is the range of improvement to be seen from lowest to highest. It is known that from **No Free Lunch Theorem** [2], there is no single algorithm that will always the best in all types of datasets. On a particular data set, one specific algorithm may work best, but some other algorithm may work better on a different data set. One good example is in **image classification problem**, deep learning algorithm has proven to have the best performance compared to traditional machine learning algorithm [1]. Another good example is in **time series forecasting problem**, the most recent comprehensive study comparing classical statistical algorithms vs machine learning algorithms including latest Neural Network algorithms [5] shows classical statistical algorithms have better performance compared to machine learning algorithms in terms of prediction accuracy for single and even for multiple horizon forecasting. Besides these examples, there is one comprehensive study done by [3] where 179 classifiers from 17 families being evaluated on the whole **UCI database (121 datasets)** which also support the No Free Lunch Theorem as well. Interestingly, from all evaluation done, it's concluded the classifiers most likely to the best are the random forest (RF) versions. Another aspect of literature review was done on any previous work done on classification problem for semiconductor manufacturing test data. There are many published articles refer to yield modeling, however most of them either uses different application or source or response. There are 2 articles which considered having similarities in terms of this work focus. The first one by [4] uses same objective. One of his presented case studies is exactly using Sort and Class data where new method being evaluated on a very small dataset (395 records with 220 input variables). The new method seems to be similar to random forest algorithm was compared to Naïve Bayes and C4.5 algorithm. The second article by [9] uses Sort and Class data as input/response variables and attempted die level prediction, which later found to be not so accurate, then used wafer level prediction with acceptable accuracy. The study only used single algorithm which is CART tree based ensemble stochastic gradient boosting. Therefore, there are 2 gaps seen, no research uses large enough sample size and in same time evaluates with high range of classifiers. This research will address the gap by using real manufacturing test data which will be selected to be similar in nature with respect of typical semiconductor Sort and Class data, the data set to be used will have more than sufficient samples (target to have > 100k of records for training), and lastly multiple algorithms to be evaluated instead of few. In addition, a new framework for model training and evaluation to be designed as well.

METHODOLOGY

The proposed methodology consists two flows in sequence. The first one is model training flow and second one is model evaluation flow. In the first flow, there are few key steps. This starts with data preparation where the selected dataset will be split to 4 non-overlapped parts. The one used for model training/testing will have higher proportion of data compared to other 3 similar sized parts and will be also partitioned based on Class test end date. The other 3 datasets will be used as validation. Next will be feature selection steps. To be specific, feature selection with redundancy elimination method [8] will be applied on input data to reduce the number of variables to very minimum. After the dataset is trimmed to only keep those important variables from feature selection step, the model training flow will have 2 classifier groupings which are control group and auto machine learning (ML) prior modeling flow. The control group will serve as reference. The other grouping, will use auto machine learning (ML) to run multiple classifiers automatically and only top 3 to be selected for next step. The performance metric used is recall rate at specified precision from ROI breakeven point. The threshold probability that correspond to fixed precision will be set as the classifier threshold during model evaluation on unseen datasets. The model evaluation flow will use 3 different non-overlapped datasets and comparison of classifiers will be based on ranking of recall rate and precision rate performance.

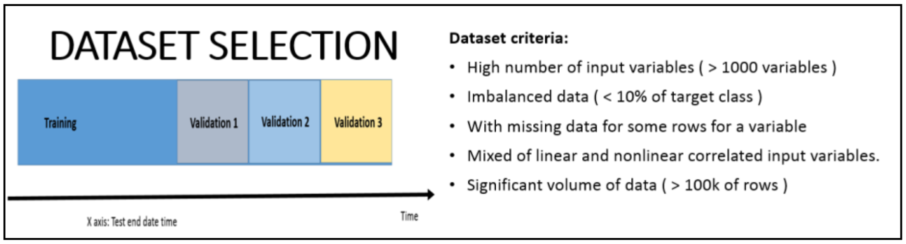
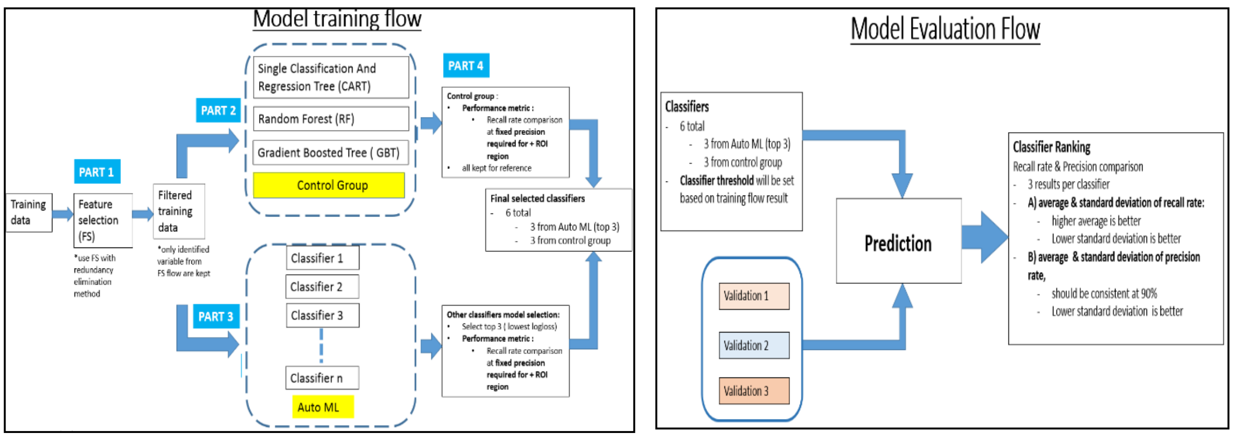


FIGURE 3. Dataset selection



(a) (b)

FIGURE 4. (a) Model Training Flow (b) Model Evaluation Flow

RESULTS

Results from the proposed flow using actual data is described as following. Dataset used has total of 5254 number of numerical variables. The training data has 170k of rows while the validation dataset has 71k per each. The next step is feature selection with redundancy elimination. Three different FS settings are used. 1) **FS default** – 561 variables identified which is equivalent to **10.6%** of original number of variables. However this is not the minimum as the algorithm will pick those relevant variables which including redundant variables as well. 2) **FS 0.01 pvalue threshold** – the number of identified variables further reduced from 561 to 351 which is equivalent to **6.6%** of original number of variables. This still includes the redundant variables as well. The number of variables selected reduced due to tighter threshold set 3) **FS + redundancy elimination** – the number of identified variables significantly reduced from 351 to 58 variables only which is equivalent to **1.1%** or the minimum list. The result from feature selection with redundancy elimination will be used in next step as proposed. For next flow in model training, 2 groups of classifiers will build models and will be compared. As mentioned previously, a min precision value is required to proceed. For this case we will use min precision of 90%. 90% precision means every 10 die screened, 9 of them are really bad die. In this stage, the control group consists of standard CART tree, Random Forest, and Gradient Boosted Tree. For Gradient Boosted Tree, 2 models used where one uses default iteration value of 50 and the other one is using 1000. Once completed, the generated model then tested on 20% unseen data from training data and result from the prediction is then used to generate precision – threshold curve. Then threshold value that crosses 90 % precision line will be set as threshold value (TV) for the classifier in model evaluation flow. The other group which uses auto machine learning (ML) has different flow. Auto ML will do auto modeling where it searches through millions of possible combinations of algorithms, preprocessing steps, features, transformations, and tuning parameters and uses supervised learning algorithms to build models and results being updated in leaderboard. There are 59 models evaluated and the top 3 (lowest error using LogLoss) are selected. The results show blended / stacking models are at top list. Then same steps were applied where threshold value (TV) being determined for each selected classifier. Finally the model evaluation flow is run, all selected classifiers from both groups which now has unique threshold value (TV) set using previous flow are tested on the 3 validation datasets. The full results from model training and model evaluation are summarized in a table 1. The full results show: Stacking models performs the best and there is average of 3.28% range from minimum to maximum on recall rate% by classifier. The predictability for these classifiers is shown by recall rate from 39.3% to 42.6%. This implies the dataset has significant underlying pattern structure. Recall rate of maximum 42.6% means 42.6% of bad die able to be predicted correctly with 90% precision. The precision for each dataset seems to be similar and maintained at ~ 90 % range.

Index	Group	Purpose	No of records	No of input variables	Type of variables
1	Training	Model training/testing	170k	5254	Numeric
2	Validation 1	Model evaluation	71k		
3	Validation 2		73k		
4	Validation 3		70k		

(a)

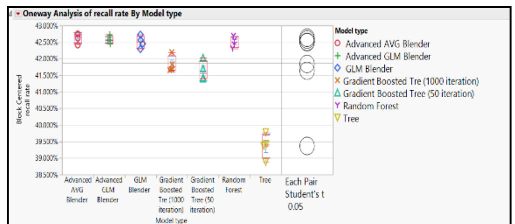
Method	Number of variables selected	Total number of variables	% of selected from total
Feature Selection (default)	561	5298	10.6
Feature Selection (0.01 p-value threshold)	351		6.6
Feature Selection (Redundancy Elimination)	58		1.1

(b)

FIGURE 5. (a) details of used dataset and its partition (b) Feature selection result summary

Flow	Dataset	Ranking	Group	Model type	Threshold Value (TV) at precision min 90%	Precision	Recall rate	Range (Max - Min)
Model training flow	training/test	1	Auto ML	Advanced AVG Blender	0.5174	90.03%	45.69%	
		2	Control	Random Forest	0.4800	90.02%	45.64%	
		3	Auto ML	Advanced GLM Blender	0.5246	90.02%	45.64%	
		4	Auto ML	GLM Blender	0.5243	90.00%	45.53%	2.83%
		5	Control	Gradient Boosted Tree (1000 iteration)	0.5630	90.04%	44.75%	
		6	Control	Gradient Boosted Tree (50 iteration)	0.5616	90.09%	44.47%	
		7	Control	Tree	0.6336	89.96%	42.86%	
Model evaluation flow	validation1	1	Auto ML	Advanced GLM Blender	[redacted]	89.97%	39.94%	
		2	Auto ML	Advanced AVG Blender		90.05%	39.89%	
		3	Control	Random Forest		89.94%	39.80%	
		4	Auto ML	GLM Blender		90.32%	39.77%	3.09%
		5	Control	Gradient Boosted Tree (1000 iteration)		88.80%	39.68%	
		6	Control	Gradient Boosted Tree (50 iteration)		89.14%	39.51%	
		7	Control	Tree		90.21%	36.85%	
	validation2	1	Auto ML	Advanced AVG Blender		90.04%	44.83%	
		2	Auto ML	Advanced GLM Blender		90.24%	44.81%	
		3	Auto ML	GLM Blender		90.28%	44.81%	
		4	Control	Random Forest		90.10%	44.51%	3.86%
		5	Control	Gradient Boosted Tree (1000 iteration)		89.02%	43.92%	
		6	Control	Gradient Boosted Tree (50 iteration)		89.03%	43.78%	
		7	Control	Tree		90.46%	40.97%	
	validation3	1	Auto ML	Advanced AVG Blender		91.80%	40.11%	
		2	Control	Random Forest		90.93%	40.06%	
		3	Auto ML	Advanced GLM Blender		91.88%	39.93%	
		4	Auto ML	GLM Blender		91.60%	39.93%	3.32%
		5	Control	Gradient Boosted Tree (1000 iteration)		90.54%	39.02%	
		6	Control	Gradient Boosted Tree (50 iteration)		90.60%	38.79%	
		7	Control	Tree		91.89%	36.79%	
							Average	3.28%

(a)



Block Dataset

Means and Std Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
Advanced AVG Blender	4	0.426323	0.001594	0.00080	0.42379	0.42886
Advanced GLM Blender	4	0.425790	0.001115	0.00056	0.42402	0.42756
GLM Blender	4	0.425093	0.001843	0.00092	0.42216	0.42802
Gradient Boosted Tre (1000 iteration)	4	0.418410	0.002495	0.00125	0.41444	0.42238
Gradient Boosted Tree (50 iteration)	4	0.416365	0.002940	0.00147	0.41169	0.42104
Random Forest	4	0.425013	0.001640	0.00082	0.42240	0.42762
Tree	4	0.393663	0.003690	0.00185	0.38779	0.39953

(b)

FIGURE 6. (a) Summary of result by grouping (b) Comparison between classifier (after dataset factor blocked)

CONCLUSION

The results in classifier training and validation on actual data sample shows the proposed framework meets the objective of this research where: 1) **Feature selection with redundancy elimination method** able to reduce number of feature list to minimum. This is required as semiconductor test data usually has hundreds to thousands of test result. The reduced list will make faster and more accurate model building and testing 2) **Threshold Value (TV) setting using minimum precision % for +positive ROI region** is proven to work and will be **novel method** which can be used for any probabilistic classifier. This will be set as post modeling step to ensure any selected classifier will maintain its precision at desired %. This eliminates the need for evaluation of different setups to get a model that meets the minimum precision which were done previously. 3) **Evaluation using Auto ML vs Control Group gives the range of opportunity for any given dataset from minimum range to maximum range of recall rate**. We can determine how much opportunity exists thus can be used as supporting data for justification on classifier selection and implementation. 4) **Evaluation results show ensemble methods give the best result** in terms of highest recall rate at fixed precision compared to single classifier. This is aligned with findings from literature review and from technical/theory perspective. Based on the results, the framework can be further simplified by removing the control group and use CART tree in auto ML group as reference.

ACKNOWLEDGMENTS

The authors would like to thank School of Quantitative Sciences, Universiti Utara Malaysia for providing the academic guidance and support in conducting this research and to thank Intel for providing assess to manufacturing test data and analytical tools.

REFERENCES

1. D. Ciresan, U. Meier and J. Schmidhuber, J, *Multi-column deep neural networks for image classification*, IEEE Conference on Computer Vision and Pattern Recognition (2012), pp. 3642-3649.
2. D. H. Wolpert, *No free lunch theorems for optimization*, IEEE transactions on Evolutionary Computation (1997), pp. 67-82.
3. M. Fernandez-Delgado, E. Cernadas, S. Barro and D. Amorim, J. Mach. Learn. Res. **15**, pp. 3133–3181 (2014).
4. L. Rokach and O. Maimon, Journal of Intelligent Manufacturing **17(3),** pp. 285-299 (2016).
5. S. Makridakis, E. Spiliotis and Y. Assimakopoulos, PLOS ONE **13(3),** pp. 1–26 (2018).
6. M. Maxfield, *2D vs. 2.5D vs. 3D ICs 101* (Retrieved from https://www.eetimes.com/document.asp?doc_id=1279540, 2012, April 8).
7. *Semiconductor Packaging History and Trends* (Retrieved from https://anysilicon.com/semiconductor-packaging-history-trends, 2016, February 12).
8. E. Tuv, E. Borisov, G. Runger and K. Torkkola, Journal of Machine Learning Research **10,** pp. 1341-1366 (2009).
9. W.K. Yip, K.G. Law and W.J. Lee, *Forecasting Final/Class Yield Based on Fabrication Process E-Test and Sort Data,* 3rd Annual IEEE Conference on Automation Science and Engineering (2009), pp. 478-483.