Privacy Preserving Amalgamated Machine Learning for Process Control

Wilfried Verachtert*[a], Thomas J. Ashby[a], Imen Chakroun[a], Roel Wuyts[a], Sayantan Das[a], Sandip Halder[a], Philippe Leray[a]
[a]imec, Kapeldreef 75, Heverlee. B-3001, Belgium

ABSTRACT

Further application of machine learning is important for the future development of semiconductor fabrication. Machine learning relies on access to large, detailed datasets. When different parts of the data are owned by different companies who do not wish to pool their data due to commercial sensitivity concerns, the benefits of machine learning can be limited resulting in reduced manufacturing performance. Imec has developed Privacy-preserving Amalgamated Machine Learning (PAML) to overcome this problem and achieve predictive performance close to models built on pooled data, without compromising sensitive raw data. In this paper we give a concrete example based on an in-house overlay metrology dataset where we apply a PAML enhanced version of a tree regression model, and quantify the performance benefit compared to separate models that don't have access to all of the data.

Keywords: Machine learning, privacy-preserving, metrology, overlay, semiconductor fabrication

1. INTRODUCTION

Semiconductor fabrication is one of humanity's most complex manufacturing processes, consisting of a large number of processing and metrology steps carried out by technologically advanced machines that require extremely fine control to be able to produce working devices. The techniques used for monitoring and control in the fab have evolved significantly over time; machine learning techniques have been used in various forms in the fab for quite some time already, and their sophistication and scope of application is growing.

The data driven models of machine learning work best when they are trained with all information available, ideally being a large number of data points described with as many relevant features as possible. Consequently, the best approach to prepare for training models is to pool all the available information. In certain contexts, non-technical issues can disallow pooling and thus restrict the availability of information that can be used for training, resulting in suboptimal models and predictions. In the great majority of cases the restriction on what can be shared is some form of privacy requirement driven by either legal protections for sensitive personal data, such as healthcare data, or the desire of companies to protect commercially sensitive information. We refer to data that cannot be universally shared for the purpose of building machine learning models as existing within a ***privacy silo***; the term is inspired by the existence of data silos in large organisations where the main barrier to combining information across data silos is an engineering one.

Semiconductor fabrication is supported by a complex ecosystem of companies developing various different types of processing and metrology tools, companies that use collections of these tools in manufacturing and various smaller companies in support roles including information management and analysis. Given the number of different companies, the complexity of the technologies, and the large amount of sensitive intellectual property involved, it is clear that attempts to apply machine learning will quickly run into non-technical barriers to pooling information if it is not already happening today.

In this paper we illustrate why privacy silos can be a problem, and show that this problem can be largely overcome by the application of Privacy-preserving Amalgamated Machine Learning. Section 2 describes the wafer processing and data collected from it. Section 3 describes the machine learning model built from the data and the results that can be achieved on pooled data. Section 4 discusses privacy silos and shows how the model accuracy goes down when data from within only one silo is available. Section 5 describes a method of building models across privacy silos and shows how it can improve model accuracy and approach the accuracy of the model built on the pooled data. Section 6 describes related work. Section 7 concludes, with acknowledgements in Section 8 and references in Section 9.

https://doi.org/10.1117/12.2584893

2. DATA

2.1 Description of the process flow -LELE approach

The process flow involves a LELE (litho-etch-litho-etch) approach to achieve a minimum metal (M1) pitch of 48nm using a 1.35NA 193nm immersion scanner. The design pattern was decomposed into two layers M1A and M1B at the same level. A simplified process flow involving the important steps is illustrated in Fig. 1. It starts with deposition of the wafer BEOL stack followed by M1A lithography exposure with immersion scanner using negative tone resist process, NTD to pattern 40nm trenches. Lithography stack was 85 nm resist coated on top of 30nm (spin on glass) SOG and 100nm (spin on carbon) SOC. After the lithography step, M1A trenches were etch transferred onto an oxide film acting as a memorization layer. A second lithography exposure was done to pattern the M1B layer. At this stage, the overlay between M1A and M1B was measured using an optical diffraction-based overlay technique (DBO); this DBO step is labeled step 4. This was followed by transferring the M1B pattern onto the same oxide memorization, or storage, layer. After that, both M1A and M1B were transferred onto a TiN hardmask (HM) layer by plasma etching. At this stage, DBO data was collected again (step 8), along with other metrology data not used in this project (CD-SEM on the electrical structures etc). The TiN HM pattern was etch transferred into a low-k dielectric material, and DBO was collected again (step 11). The full M1 trenches patterned in the low-k dielectric layer were filled up with Cu and CMP'd (chemical mechanical polishing), after which a final round of DBO was collected, labeled step 15 (followed by other metrology measurements). A 5nm thin layer of SiCN was also deposited on top of the wafer to prevent oxidation of Cu metal lines. Electrical capacitance measurements were made on the fork structures after processing.

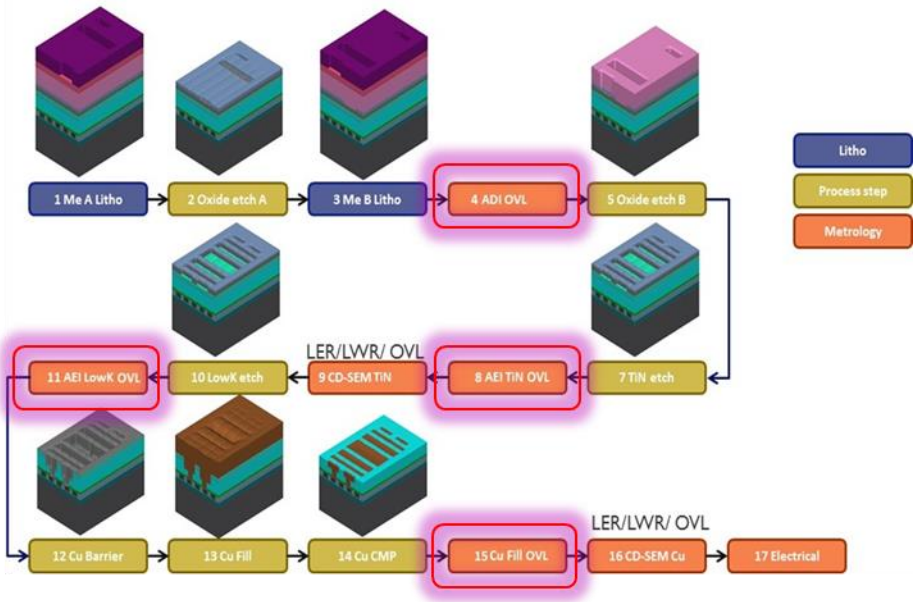

Figure 1: Process flow showing LELE approach to achieve 48nm pitch. The highlighted steps are those that generate the optical overlay measurement data used in this paper.

2.2 Description of the electrical test structures

Fig.2 describes the 12 vertically placed and 12 horizontally placed (in the layout) fork-fork structures to measure capacitance. Due to overlay errors, the dielectric distance between M1A and M1B changes and can be directly correlated with X overlay for vertical lines and with Y overlay values for horizontal lines. The design CD here is 24nm and distance

https://doi.org/10.1117/12.2584893

between the two metal lines changes in steps of 2nm. For example, the distance between M1A and M1B is 24nm for the AB6 and 34 nm for AB1. Similarly, it is 24nm for BA1 and 34nm for BA6.

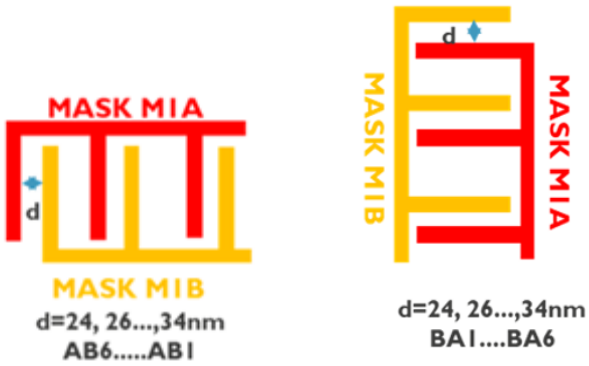

Figure 2: Description of capacitance measurement structure

The target data to predict used in the experiments reported here is the electrical capacitance measurements of the forks with vertically placed lines with the separation of 30 nm, located at sub-die site zero (being one measurement per die).

2.3 Wafers and Programmed Overlay

Four wafers were processed using this flow, each containing 128 dies. Two of the wafers were created with programmed overlay by creating a scanner sub-recipe such that wafers receive a translational offset increasing from 0 to ±7.5nm in X and Y direction in four of the columns of dies on the wafer. A visualisation of DBO measurements for a non-programmed overlay and a programmed overlay wafer are shown in Fig. 3 to give an indication of how programmed overlay is introduced.

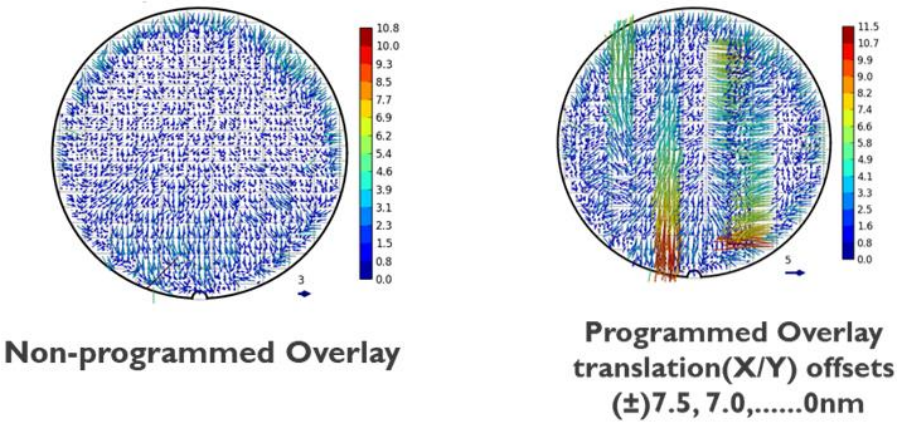

Figure 3: Overlay data plotted on a wafer map to show where programmed overlay is introduced. On the left is a normal wafer, on the right is a wafer with programmed overlay.

https://doi.org/10.1117/12.2584893

2.4 DBO measurements used

The features being used to predict the electrical measurements are the DBO measurements. Predictive models are built using data from either one (the local within-privacy-silo models) or all four steps (either the global or the PAML models) where DBO measurements are taken. Each set of DBO measurements consists of one or more measurements of multiple optical targets per die. Each optical measurement can vary according to optical settings such as the dose, polarization and wavelength, with some steps consisting of multiple DBO measurements with different settings.

3. MACHINE LEARNING FOR ELECTRICAL PROPERTIES

3.1 Aim

Our initial aim for the machine learning is to show that the electrical measurements can be predicted from the DBO measurements made at earlier steps in the process. Of the available DBO measurements, an initial assessment was made to identify the informative ones, and on that basis 8 measurements were selected for each DBO step. In total this gives 32 feature measurements per die for the whole data set. On this basis, a regression model is built to predict the single electrical target measurement per die, based on the 32 features for that die.

All die locations are treated equally, in that the data set doesn't identify the die location within a wafer neither for training nor test.

3.2 Method

The data set is divided into a training set of 3 wafers and a hold-out test set of 1 wafer. With 128 dies per wafer, the total size of the training set is 384 and the total size of the test set is 128. As the training set is relatively small, data augmentation is performed by replicating the test set 3 times. The data was also normalised on a per feature basis, by centering and scaling with respect to the maximum absolute value. The target data to predict was also normalised in the same way.

A regression model was built using the ExtraTreeRegressor [1] from the SciKit Learn machine learning package [2]. Due to time limits we didn't perform an extensive hyperparameter search, but chose a forest size of 64 and a minimum sample split size of 16 (with a view to reducing the risk of overfitting), with all other parameters left at their default settings.

The model that was built was assessed by looking at the error on the dies in the hold-out wafer using the R2 regression score.

3.3 Results

Fig. 4 shows the plot of actual values vs predicted values for the electrical measurement on the test set for a subset of the values (bottom right). The overall score R2 is 0.98 (with the maximum score being 1.0). Next to the plot of values, there is a wafer map showing the total error of the prediction. Above, there is a wafer map that shows on a colour-intensity scale the actual value in the surrounding box and the predicted value in the central circle.

https://doi.org/10.1117/12.2584893

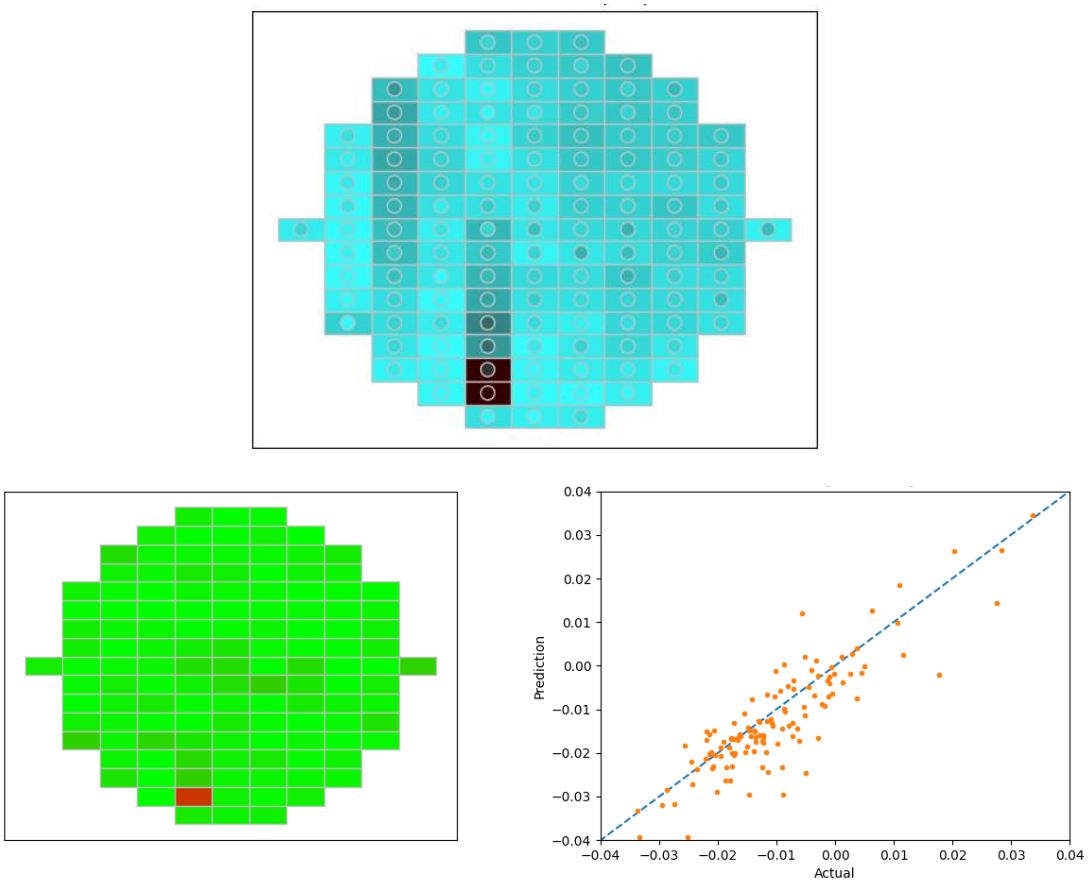

Figure 4 Global model *Top*: Wafer map showing per die the actual value (surrounding colour) and the predicted value (colour in the central circle). *Bottom left*: Overall error for predictions shown per die (light green is low error, red is high error). *Bottom right*: Plot showing prediction *vs* actual normalised value for a selection of the dies.

4. PRIVACY SILOED MODELS AND INFORMATION LOSS

The amount of data generated in one form or another in a modern fab is huge [3]. However, the data that is generated is by no means universally accessible. Data generated by fab processing and metrology tools is usually considered by fab operators to be sensitive, especially if the data can be used to gain insight into recipes used at the fab, or if there is potential to leak commercially sensitive aspects of fab operation such as yield and processing throughput. Consequently, it is by no means guaranteed that fab operators are willing to share tool output even with the vendors of the tools installed in their fab.

Similarly, fab tools themselves generate a large amount of information during their operation. However, this output is usually condensed before being delivered to tool operators in the form of tool logs or metrology output. For example, there are now metrology tools on the market that rely internally on machine learning models in order to produce their measurements, where the exact details of the machine learning model and the data it is built from are not public and are presumably considered commercially sensitive.

Furthermore, although fab operators have access to the tool output for the tools that they own in their fabs, the direct flow of information between tools from different vendors within a fab is often non-existent, with the net result being that the fab operator is the only one that can build models of the interaction between the two tools, and the possibility to steer their interaction is limited.

Consequently, one can view the data generated in fabs as existing in privacy silos. Each actor within the fab ecosystem (fab operator, processing tool vendor ***A***, processing tool vendor ***B***, metrology tool vendor ***C*** et cetera) has access to data

https://doi.org/10.1117/12.2584893

within their own privacy silo, but not the data in the other privacy silos (although there will be overlap between some silos, with the fab operator probably having access to the largest volume of information).

The existence of privacy silos is likely to have a negative impact on the quality of machine learning models that can be built to improve various aspects of fab operation, on the basis that machine learning models improve with access to larger amounts of information. However, quantifying this is difficult without having a total overview of all available data in the fab, which is difficult to get due to the lack of public information on exactly what data is generated inside each tool, let alone getting access to that data. Consequently, we have tried to illustrate the negative effects of privacy silos with some illustration-of-concept experiments based on the dataset and machine learning model described in sections 2 and 3.

Table 1. R2 regression scores for local ETR models and the global model built on pooled information

Model	R2 Score
Local Step 4	0.90
Local Step 8	0.95
Local Step 11	0.95
Local Step 15	0.95
Global model	0.98

4.1 Method

We repeated the experiments from section 3, but this time building four different models, each one trained only on the DBO features available at a particular step of the processing flow. Thus, each model has only eight features for each data point. This mimics the set-up of four different privacy silos where data is only available locally within each one.

4.2 Results

The R2 scores resulting from the models trained at each step are given in Table 1. As can be seen from the table, the models built on the features from the individual steps do less well than the model with the global overview. This is shown graphically in Fig. 5, where the prediction is shown against the actual value, and an indication of prediction quality is given, for the four local models built using information only available at a single step.

https://doi.org/10.1117/12.2584893

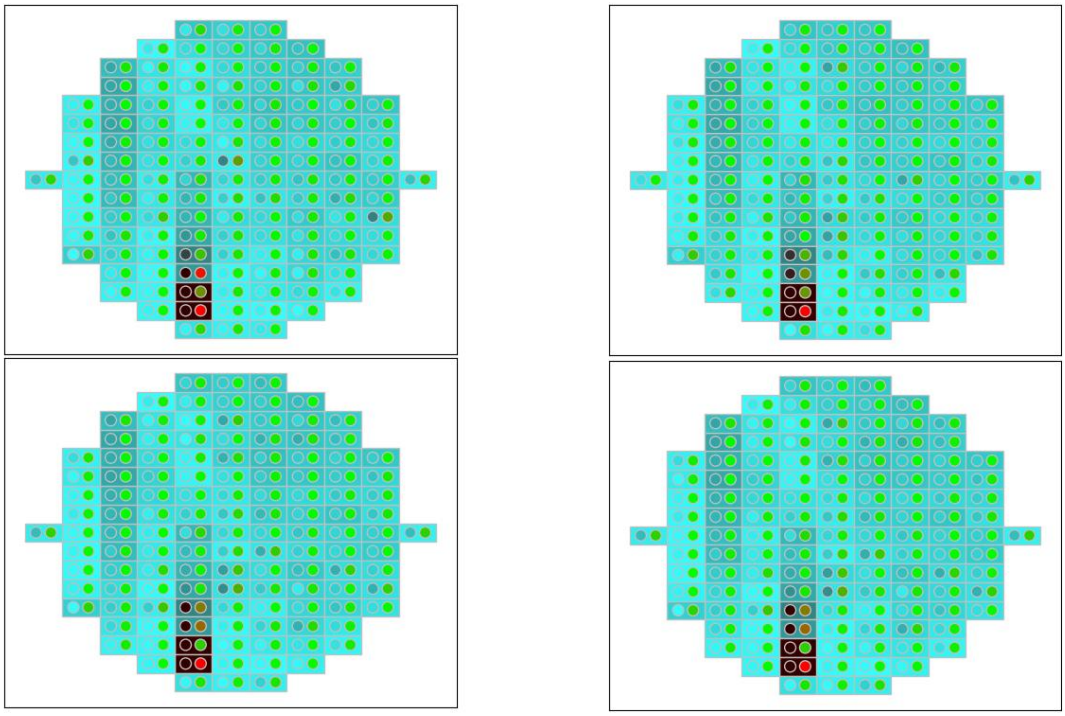

Figure 5 Wafer map plots showing the actual value for a die (surrounding box) the value predicted by the local model (left circle) and the quality of the prediction (red-green scale, right circle). *Top Left*: Step 4. *Top Right*: Step 8. *Bottom Left*: Step 11. *Bottom Right*: Step 15

5. PRIVACY-PRESERVING AMALGAMATED MACHINE LEARNING (*PAML*)

The reasons for the existence of privacy silos make it impossible to simply pool all available data as a precursor to building a model. In order to mitigate the problems caused by privacy silos, Imec has developed *Privacy-Preserving Amalgamated Machine Learning* (PAML), a patented [4] approach to building machine learning models without the need to pool data by carefully separating out what each partner in the process can see to prevent the sharing of sensitive data whilst improving prediction results.

5.1 PAML-trees

For this experiment, the Extra Trees Regressor (ETR) was adapted using the principles of PAML resulting in the PAML Extra Trees Regressor. The first step in a PETR model is making the local ETR models for the different steps, each within their own privacy silo, identically to section 4.1. The second step is to extract suitably obscured information, called *PAML features*, from all of the local models which can be safely shared outside of the privacy silo. The third step is to construct a new PAML model within each silo, with overall improved predictions with respect to the local models through use of the PAML features.

For PETR, the PAML features of a given local model are derived from its ensembled trees. Each tree is a regression tree, and each of its leaf nodes can be assigned a label (e.g. a letter A, B or C etc) that distinguishes it from the other leaves, whilst revealing very little about the structure of the tree. Any given data point to predict an output for will follow a path through the tree based on its features, ending up at a final node that determines the output for that tree. Thus, the sample can be characterised to some extent by the collection of labels (one per tree) that it generates in the ensembled forest. These labels, or a subset of them, are the PAML features for that sample on that local model. Note that the PAML features reveal nothing directly about the values of the measured features for the associated data point, and hence can be shared outside

https://doi.org/10.1117/12.2584893

of the privacy silo without leaking any valuable information. The collection of PAML features for a data point is the union of the PAML features from each local model.

The construction of the final PETR model within each privacy silo is relatively straightforward. The model-building procedure is repeated using the local data, but now enhanced with the collection of PAML features as inputs for each data point.

After building the model, the predictions on new data are generated by first creating the PAML features using the local models, pooling and sharing the PAML features, and then making the final prediction using the PAML-feature enhanced model.

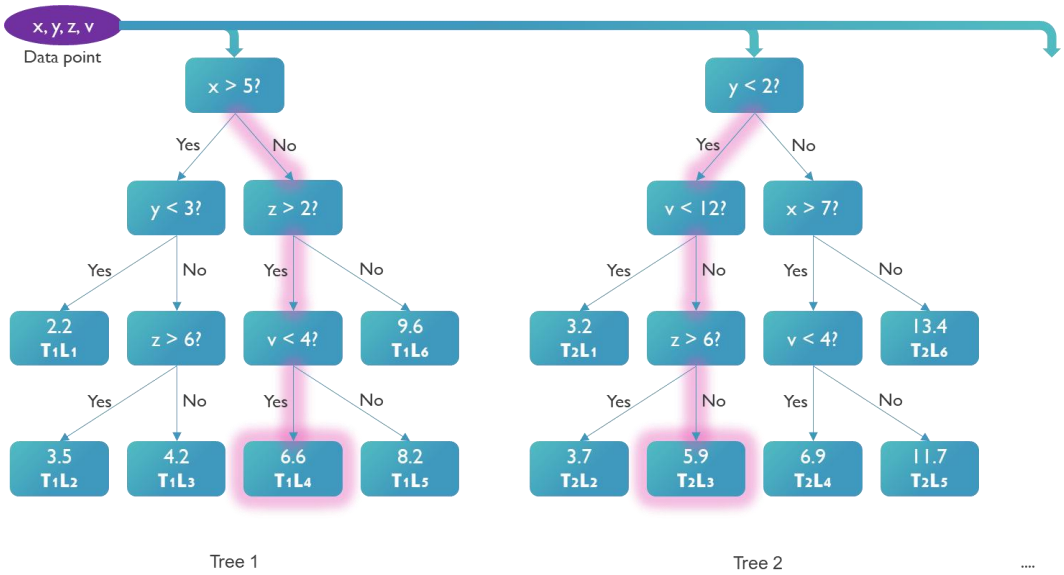

Figure 4: Diagram showing how terminal nodes in a decision tree are labelled to produce PAML features. The number above is the prediction value produced by that particular tree in the local-only model within the privacy silo; these numbers output by the tree ensemble are averaged together to give the final prediction. The string below is the value of the PAML feature for that terminal node in that tree. In the example shown, the data point produces the first to entries in the PAML feature vector as (T1L4, T2L3, ...). Further trees are omitted for space reasons. Note that unlike the numerical values, the PAML feature strings aren't collapsed together.

5.2 Results

Figure 7 shows the prediction results from the PETR models. These R2 scores are tabulated for comparison in Table 2. The results show an improved accuracy for the PAML models vs the local models, with R2 approaching that of the global model for the PAML models. This is shown graphically in Fig. 7 in a similar fashion to Fig. 5; the prediction is shown against the actual value, and an indication of prediction quality is given, for the four PAML models built using information available at a single step augmented with the PAML features.

https://doi.org/10.1117/12.2584893

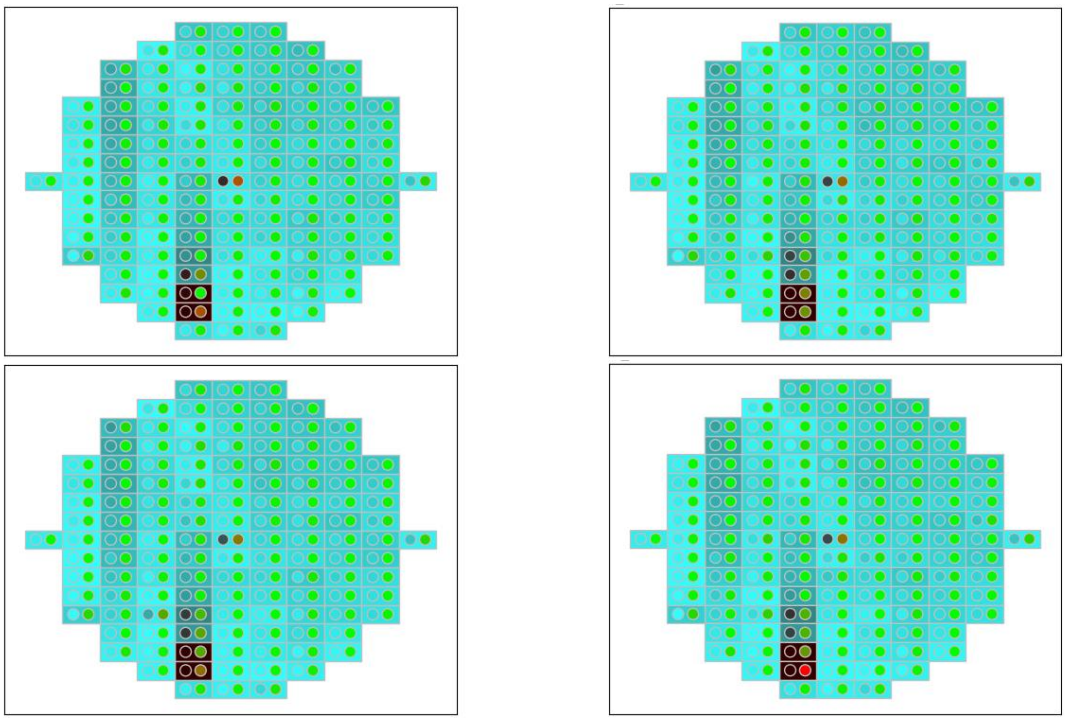

Figure 7 Wafer map plots showing the actual value for a die (surrounding box) the value predicted by the PAML model (left circle) and the quality of the prediction (red-green scale, right circle). *Top Left*: Step 4. *Top Right*: Step 8. *Bottom Left*: Step 11. *Bottom Right*: Step 15

Table 2. R2 regression scores for local ETR models, PETR models and the global model built on pooled information

Model	R2 Score local ETR	R2 Score PETR (PAML)
Step 4	0.90	0.97
Step 8	0.95	0.97
Step 11	0.95	0.97
Step 15	0.95	0.95
Global model	-	0.98

5.3 Discussion

Although this example for the use of PAML is built using a dataset that does not intrinsically consist of privacy silos, and thus may seem somewhat artificial, this is due to a problem of circularity; to convince potential partners to give access to their privacy silos we need to first show publicly that a technique that deals with privacy silos can work. Hence this approach of an illustration-of-concept, using a real data set with a contrived separation of information.

The description of PETR demonstrates the general principles of PAML applied to regression tress. The PAML framework can be applied to various classifier and regression models to produce privacy respecting variants.

6. RELATED WORK

https://doi.org/10.1117/12.2584893

6.1 Machine Learning in the Fab

Of all industrial production processes, the most research on the application of machine learning has been done for electronics manufacturing [5]. ML applied to various aspects of semiconductor device manufacturing forms a sizeable chunk of this work, and has a long history with scientific articles stretching back around 30 years [6].

Modern manufacturing of semiconductor devices is an extremely complex process, with wafers passing through many repeated steps from different families of processing step including lithography, deposition, planarization and etching steps. Interspersed between the steps are various metrology steps that are used to monitor and control the process. Semiconductor manufacturing as a whole does not directly match with the domain of classical machine learning in that the data generation process is not purely passive; in most machine learning settings, the process generating the data set is assumed to be outside the control of the person who is building the ML model, relegating them to the role of passive observer. By contrast, production processes have numerous parameter settings that can in theory be changed by the person building a model, and consequently aspects of model building fit more naturally in the domain of design of experiments. However, once the production process has been set up the chosen parameters are essentially fixed, at which point the generation of data is much closer to a classical ML setting. Away from the choosing of machine parameters, there are clearly aspects of the production process where data generation is not controlled by the engineers. These include detecting when things go wrong in the processing and handling of wafers, predicting when maintenance needs to be carried out on tools, extrapolating metrology measurements to unmeasured areas of the wafer, dealing with process drift, and understanding the root causes of processing problems when then occur. Other topics covered include mitigating noise in metrology measurements, and spotting and classifying defects on wafers.

Of these topics, four that are fairly localised within the production chain have been discussed by the community for a long time [7]. The topic of virtual metrology (VM) deals with taking actual sparse metrology measurements and optionally information from processing tools and predicting missing metrology measurements, or in extreme cases skipping metrology entirely and making predictions based only on processing tool output. The usual aim is to avoid the cost, throughput and latency problems of actual metrology. Recent research has included using more sophisticated machine learning models [8] and dealing with drift [9][10]. Fault detection and classification (FDC) systems try to detect and identify processing errors occurring in individual tools. Run-to-run control (R2R) deals with limited process drift within an individual tool. Predictive maintenance (PdM) attempts to reduce maintenance costs with respect to scheduled maintenance by only taking tools off-line when it is absolutely necessary [11]. Of these four, the first three seem to be well established in industrial practice, albeit often in a limited way (concentrating on individual measurements or tools and using univariate rather than multivariate modelling, and with less advanced machine learning techniques).

Root cause analysis (RCA) within fabs is a challenging topic that has also been under consideration for a long time (e.g. [6][12][13]), but which concerns the whole manufacturing process. The ultimate aim is to improve processing by understanding which aspect of processing is driving particular problems for poor wafers, usually formulated as improving yield. RCA is challenging because poor wafers occur rarely, meaning that there is little information on which to build RCA models, which is further exacerbated by the fact that many tools and parameters and the interactions between them might be responsible for causing yield loss. Consequently, performing RCA in practice requires significant expert knowledge and time. Recent works in RCA have included more sophisticated techniques such as deep neural networks to analyse spatial patterns of faults and link them to RCA [14][15], extraction of higher-order features from convolutional neural networks on sensor timeseries data to aid RCA [16].

As feature sizes shrink, the problem of signal-to-noise ratio in metrology gets worse. This has led to research on de-noising of SEM images [17] and automatic identification and classification of defects [18].

6.2 Privacy-Preserving Machine Learning

PP-ML is a large field. The topic can be subdivided into different subfields based on the main approaches taken by the authors, with the three main approaches being secure multi-party computation (SMPC), homomorphic encryption (HE), and differential privacy (DP).

The subfield of DP, and the related notion of compressive privacy [19], are based on perturbing the data. If the data is sufficiently altered (e.g. enough noise is added to the data), then it becomes difficult to figure out what the original

https://doi.org/10.1117/12.2584893

underlying data is, thus protecting it [20]. Once it is protected, the data can be shared publicly thus allowing pooling of the data to learn a model. It is of course necessary to find a level of noise that provides acceptable privacy whilst degrading the accuracy of models on that data as little as possible [21], which can be difficult [22]. DP has the advantage of being agnostic with respect to the learning method used as only the data is altered.

HE is based on encrypting a private dataset, and subsequently operating directly on the encrypted data [23]. The usual aim of HE for ML is to allow the learning phase to be outsourced to an un-trusted partner. The main disadvantage of HE is that operating directly on encrypted data makes computational operations much more expensive than those done on unencrypted data [24]. This can severely hamper the original aim in that the price saving of outsourcing the learning phase may end up being less than the extra cost of performing HE operations on the outsourced platform. In addition, HE is often in practice limited to simple arithmetic operations and fixed width datatypes due to the excessive cost of multiplication and the complexities of dealing with floating point, which makes it challenging to implement full training algorithms [25].

SMPC as a subfield is itself diverse. SMPC typically deals with multiple data providers wish to keep their data secret and yet perform some operation on the totality of the data and share the result. Various works on SMPC deal with relatively simple operations that can be performed in this way, and then proceed to build training algorithms for particular ML models on top of these building blocks [26][27][28]. SMPC can be applied to parts of an algorithm, such as a model update step that combines locally computed updates [29]. SMPC approaches, similarly to HE, can require significant extra computational and communication effort above that required for locally training a model.

In addition to the three principal subfields given above, other important crosscutting topics include federated learning (FL), block chain (BC) and trusted computing platforms (TCP). FL is chiefly concerned with learning models in an environment where data cannot be brought together [30], often due to the practicalities around data transfer for huge datasets. Because data is not pooled, it often has many similarities with SMPC, but often without an explicit emphasis or strong guarantees on privacy. BC offers a means of recording actions that take part during the distributed training of a ML model, and thus can be used to help guard against deliberate poisoning of the process [31], but on its own does not add much in the way of privacy. TCP is a hardware centric solution to enable sensitive data to be handled inside protected enclaves on a remote server [32] and can provide very efficient implementations of machine learning; however, it fundamentally relies on trusting the provider of the TCP and so doesn't preserve privacy in an absolute sense in that if there is a bug in the TCP hardware design then the data may be compromised.

The work in this paper deals with a PP ML extension of a specific ML algorithm, data that is held locally by multiple parties and not revealed, and an adaptation of an aggregation step to allow suitably obscured information derived from local data to be used by other un-trusted parties. As such, it is closest to work in the area of SMPC. However, it does not rely on the usual building blocks of SMPC that compute some output from inputs that have to be kept secret, but rather shares the PAML features directly.

6.3 Privacy-Preserving Machine Learning in the Fab

To the best of our knowledge, this is the first publication directly looking at applications of PP-ML for semiconductor manufacturing.

7. CONCLUSION AND FUTURE WORK

In this paper we have given an example of machine learning being used to solve prediction problems related to semiconductor processing and metrology, where the example is based on optical overlay measurements and electrical test structure properties. We have discussed the general problem of privacy silos in the context of the semiconductor fab eco-system, and how they can reduce the effectiveness of machine learning by limiting the information available for model training. To tackle this problem we have introduced the framework of Privacy-preserving Amalgamated Machine Learning (PAML) that provides bridging between privacy silos in a way that doesn't leak sensitive information and enables the construction of machine learning models that are better than purely local within-silo models and approach the performance of global models resulting from the pooling of all data without respecting privacy constraints. Finally we have given a concrete instance of the PAML approach applied to Extra Tree Regressors, resulting in PAML Extra Tree Regressors, and

https://doi.org/10.1117/12.2584893

provided experimental results that show that PAML can regain the majority of the predictive performance loss induced by privacy silos.

Our main avenue for further investigation is to apply PAML to fab data in current fab privacy silos to further improve the results obtainable with machine learning. Working with partners to understand existing privacy silos will be a major part of this.

8. ACKNOWLEDGMENT

This project has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 826589. This Joint Undertaking receives support from the European Unions Horizon 2020 research and innovation programme and Netherlands, France, Italy, Belgium, Germany, Austria, Hungary, Romania, Sweden and Israel.

This research also received funding from the Flemish regional government (AI Research Program).

9. REFERENCES

[1] P. Geurts, D. Ernst., and L. Wehenkel, Extremely randomized trees, Machine Learning, 63(1), 3-42, 2006.
[2] Pedregosa, F, et al, Scikit-learn: Machine Learning in Python, Journal of Machine Learning Research, volume 12, pages 2825-2830, 2011
[3] Moyne J, Iskandar J. Big Data Analytics for Smart Manufacturing: Case Studies in Semiconductor Manufacturing. Processes. 2017; 5(3):39. https://doi.org/10.3390/pr5030039
[4] H Ceulemans, R Wuyts, W Verachtert, J Simm, A Arany, Y Moreau, C Herzeel. Secure Broker-Mediated Data Analysis and Prediction. US Patent App. 15/722,742, 2019.
[5] Ziqiu Kang, Cagatay Catal, Bedir Tekinerdogan, Machine learning applications in production lines: A systematic literature review, Computers & Industrial Engineering,Volume 149, 2020, 106773, ISSN 0360-8352, https://doi.org/10.1016/j.cie.2020.106773.
[6] K. B. Irani, J. Cheng, U. M. Fayyad and Z. Qian, Applying machine learning to semiconductor manufacturing, in *IEEE Expert*, vol. 8, no. 1, pp. 41-47, Feb. 1993, doi: 10.1109/64.193054.
[7] Susto, G. A., Pampuri, S., Schirru, A., De Nicolao, G., & McLoone, S. (2012). Automatic Control and Machine Learning for Semiconductor Manufacturing: Review and Challenges. Paper presented at 10th European Workshop on Advanced Control and Diagnosis 2012, Copenhagen, Denmark.
[8] M. Maggipinto, M. Terzi, C. Masiero, A. Beghi and G. A. Susto, A Computer Vision-Inspired Deep Learning Architecture for Virtual Metrology Modeling With 2-Dimensional Data, in IEEE Transactions on Semiconductor Manufacturing, vol. 31, no. 3, pp. 376-384, Aug. 2018, doi: 10.1109/TSM.2018.2849206.
[9] M. Azamfar, X. Li and J. Lee, Deep Learning-Based Domain Adaptation Method for Fault Diagnosis in Semiconductor Manufacturing, in IEEE Transactions on Semiconductor Manufacturing, vol. 33, no. 3, pp. 445-453, Aug. 2020, doi: 10.1109/TSM.2020.2995548.
[10] J. Feng, X. Jia, F. Zhu, J. Moyne, J. Iskandar and J. Lee, An Online Virtual Metrology Model With Sample Selection for the Tracking of Dynamic Manufacturing Processes With Slow Drift, in IEEE Transactions on Semiconductor Manufacturing, vol. 32, no. 4, pp. 574-582, Nov. 2019, doi: 10.1109/TSM.2019.2942768.
[11] Jalali, A., Heistracher, C., Schindler, A., Haslhofer, B., Nemeth, T., Glawar, R., Sihn, W., & Boer, P.D. (2019). Predicting Time-to-Failure of Plasma Etching Equipment using Machine Learning. 2019 IEEE International Conference on Prognostics and Health Management *(ICPHM)*, 1-8
[12] Turney, P.D. (1995). Data Engineering for the Analysis of Semiconductor Manufacturing Data. IJCAI-95 Workshop on Data Engineering for Inductive Learning
[13] McCann, M., Li, Y., Maguire, L., & Johnston, A. (2010). Causality Challenge: Benchmarking relevant signal components for effective monitoring and process control. NIPS Causality: Objectives and Assessment.
[14] K. Nakata, R. Orihara, Y. Mizuoka and K. Takagi, "A Comprehensive Big-Data-Based Monitoring System for Yield Enhancement in Semiconductor Manufacturing," in *IEEE Transactions on Semiconductor Manufacturing*, vol. 30, no. 4, pp. 339-344, Nov. 2017, doi: 10.1109/TSM.2017.2753251.

https://doi.org/10.1117/12.2584893

[15] G. Tello, O. Y. Al-Jarrah, P. D. Yoo, Y. Al-Hammadi, S. Muhaidat and U. Lee, Deep-Structured Machine Learning Model for the Recognition of Mixed-Defect Patterns in Semiconductor Fabrication Processes, in *IEEE* Transactions on Semiconductor Manufacturing, vol. 31, no. 2, pp. 315-322, May 2018, doi: 10.1109/TSM.2018.2825482.
[16] K. B. Lee, S. Cheon and C. O. Kim, A Convolutional Neural Network for Fault Classification and Diagnosis in Semiconductor Manufacturing Processes, in IEEE Transactions on Semiconductor Manufacturing, vol. 30, no. 2, pp. 135-142, May 2017, doi: 10.1109/TSM.2017.2676245.
[17] Dorin Cerbu, Sandip Halder, and Philippe Leray, Deep-learning-based SEM image denoiser (Conference Presentation), Proc. SPIE 10959, Metrology, Inspection, and Process Control for Microlithography XXXIII, 1095916 (26 March 2019); https://doi.org/10.1117/12.2515182
[18] S. Cheon, H. Lee, C. O. Kim and S. H. Lee, Convolutional Neural Network for Wafer Surface Defect Classification and the Detection of Unknown Defect Class, in IEEE Transactions on Semiconductor Manufacturing, vol. 32, no. 2, pp. 163-170, May 2019, doi: 10.1109/TSM.2019.2902657.
[19] T. Chanyaswad, J. M. Chang and S. Y. Kung, A compressive multi-kernel method for privacy-preserving machine learning, 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, 2017, pp. 4079-4086. doi: 10.1109/IJCNN.2017.7966371
[20] Cynthia Dwork; Aaron Roth, The Algorithmic Foundations of Differential Privacy, 2014, doi: 10.1561/0400000042.
[21] Kato Mivule, Claude Turner, Soo-Yeon Ji, Towards A Differential Privacy and Utility Preserving Machine Learning Classifier, Procedia Computer Science, Volume 12, 2012, Pages 176-181, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2012.09.050.
[22] Jayaraman B, Evans D, Evaluating Differentially Private Machine Learning in Practice, 28th USENIX Security Symposium (USENIX Security 19) ISBN Number 978-1-939133-06-9
[23] Jaydip Sen. Homomorphic Encryption — Theory and Application, Theory and Practice of Cryptography and Network Security Protocols and Technologies, IntechOpen, DOI: 10.5772/56687
[24] C. Moore, M. O'Neill, E. O'Sullivan, Y. Doröz and B. Sunar, Practical homomorphic encryption: A survey, 2014 IEEE International Symposium on Circuits and Systems (ISCAS), Melbourne VIC, 2014, pp. 2792-2795, doi: 10.1109/ISCAS.2014.6865753.
[25] Louis J. M. Aslett and Pedro M. Esperança and Chris C. Holmes, Encrypted statistical machine learning: new privacy preserving methods, 2015, 1508.06845 arXiv (stat.ML)
[26] Rahul Rachuri and Ajith Suresh, Trident: Efficient 4PC Framework for Privacy Preserving Machine Learning, 2019, 1912.02631 arXiv (cs.LG)
[27] Byali, M., Chaudhari, H., Patra, A., & Suresh, A. (2020). FLASH: Fast and Robust Framework for Privacy-preserving Machine Learning, Proceedings on Privacy Enhancing Technologies, 2020(2), 459-480. doi: https://doi.org/10.2478/popets-2020-0036
[28] Koti, N., Pancholi, M., Patra, A., & Suresh, A. (2020). SWIFT: Super-fast and Robust Privacy-Preserving Machine Learning. IACR Cryptol. ePrint Arch., 2020, 592.
[29] K. Xu, H. Yue, L. Guo, Y. Guo and Y. Fang, Privacy-Preserving Machine Learning Algorithms for Big Data Systems, 2015 IEEE 35th International Conference on Distributed Computing Systems, Columbus, OH, 2015, pp. 318-327, doi: 10.1109/ICDCS.2015.40.
[30] M. Aledhari, R. Razzak, R. M. Parizi and F. Saeed, Federated Learning: A Survey on Enabling Technologies, Protocols, and Applications, in IEEE Access, vol. 8, pp. 140699-140725, 2020, doi: 10.1109/ACCESS.2020.3013541.
[31] H. Kim, S. Kim, J. Y. Hwang and C. Seo, Efficient Privacy-Preserving Machine Learning for Blockchain Network, in IEEE Access, vol. 7, pp. 136481-136495, 2019, doi: 10.1109/ACCESS.2019.2940052
[32] Schellekens, D. (2013). Design and Analysis of Trusted Computing Platforms, KU Leuven PhD Thesis

https://doi.org/10.1117/12.2584893