Multiview Learning with Missing Views and Learning Solutions for Cross-Process Modeling in Semiconductor Manufacturing Industry

Anastasiia Doinychko

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : MSTII - Mathématiques, Sciences et technologies de l'information, Informatique
Spécialité : Mathématiques et Informatique
Unité de recherche : Laboratoire d'Informatique de Grenoble

L'apprentissage multivue avec vues manquantes et solutions d'apprentissage pour la modélisation inter-process dans l'industrie des semi-conducteurs

Multiview Learning with Missing Views and Learning Solutions for Cross-Process Modeling in Semiconductor Manufacturing Industry

Présentée par :

Anastasiia DOINYCHKO

Direction de thèse :

Massih-Reza AMINI Professeur des Universités, Université Grenoble Alpes	Directeur de thèse
Andres TORRES Mentor Graphics	Co-encadrant de thèse

Rapporteurs :

PUNEET GUPTA
Professeur, University of California, Los Angeles
GAËL DIAS
Professeur des Universités, UNIVERSITE DE CAEN NORMANDIE

Thèse soutenue publiquement le **6 février 2023**, devant le jury composé de :

MASSIH-REZA AMINI Professeur des Universités, UNIVERSITE GRENOBLE ALPES	Directeur de thèse
PUNEET GUPTA Professeur, University of California, Los Angeles	Rapporteur
GAËL DIAS Professeur des Universités, UNIVERSITE DE CAEN NORMANDIE	Rapporteur
GEORGES QUENOT Directeur de recherche, CNRS DELEGATION ALPES	Président
DOMINIQUE VAUFREYDAZ Maître de conférences HDR, UNIVERSITE GRENOBLE ALPES	Examinateur
MARTA SOARE Maître de conférences, UNIVERSITE D'ORLEANS	Examinatrice

Résumé

Le contrôle avancé des procédés (ou Advanced Process Control - APC en anglais) est une direction de recherche dans l'industrie de la fabrication de semi-conducteurs (ou Semiconductor Manufacturing - SM en anglais) engagée dans le développement de diagnostics de processus automobiles et de solutions de gestion de produits pour préserver un rendement élevé en fin de ligne et réduire le risque de défaillance de l'équipement.

Dans cette thèse, nous nous concentrons sur l'étude des approches basées sur l'apprentissage machine (ou Machine Learning en ML) pour développer un cadre unifié pour l'analyse et la modélisation des processus à partir de données SM multi-vues très diverses (provenant de différentes sources).

Les fonctionnalités multi-vues globales donnent une description plus complète d'un phénomène, et l'apprentissage multi-vues est généralement mieux adapté que l'apprentissage mono-vue (ou à vue unique), ce qui motive ce travail. L'un des principaux défis ouverts dans le domaine APC dans SM est la capacité à tirer parti de la richesse des informations pour caractériser pleinement le processus et déterminer la valeur des nouvelles métriques. Dans cette thèse, nous analysons les techniques de traitement de données existantes et nous exposons une stratégie qui consiste en des étapes de nettoyage des données, d'extraction des caractéristiques et de sélection de variables pour faire face aux imperfections des données;telles que le bruit, les étapes d'échantillonnage irrégulières dans les données de séries chronologiques sensorielles et les enregistrements incomplets, tous dus au taux d'erreur de corruption naturelle des outils d'enregistrement.

Cette thèse vise également à élargir le champ de la modélisation des processus traditionnels en SM grâce à l'analyse inter-processus. La fabrication du produit est une procédure séquentielle d'application de processus ordonnés pour déposer de nouvelles couches de fonctionnalité qui permettent d'utiliser l'historique des précédents pour connaître son impact sur la cible de modélisation actuelle d'intérêt. Dans ce sens, nous proposons une méthodologie qui bénéficie non seulement de différents types de mesures, mais également des dépendances entre les différentes étapes du processus pour rendre les processus plus prévisibles et productifs.

De plus, nous étudions le problème des données manquantes, principalement lorsqu'une des vues est manquante, ce qui est un autre défi ouvert dans le domaine de l'apprentissage. Certaines études abordent ce problème en supposant l'existence de fonctions de génération de vues pour compléter approximativement les vues manquantes. Cependant, ces fonctions nécessitent généralement une ressource externe pour être définies, et leur qualité impacte directement les performances du modèle prédictif final appris sur l'ensemble d'apprentissage terminé. Au lieu de cela, dans ce travail, nous proposons d'aborder ce problème en apprenant

conjointement les vues manquantes et l'estimateur cible multi-vues en utilisant une approche d'apprentissage antagoniste inspirée par la capacité des réseaux antagonistes génératifs (ou Generative Adversarial Netwotks - GAN en anglais) à saisir la distribution sous-jacente des données et créer de nouveaux échantillons.

Finalement, nous considérons les tâches APC telles que la métrologie virtuelle et la maintenance prédictive pour mener des expériences en utilisant les collections de données réelles fournies par les principales compagnies de fabrication de fabrication de semi-conducteurs en Europe avec lesquelles nous avons collaboré, dans le cadre du projet "Metrology Advances for Digitized Electronic Components and Systems (ECS) Industry 4.0 (MADEin4)". De plus, étant donné que le problème des données manquantes dans les collections multi-vues est répandu dans différents ensembles de données au-delà de l'industrie SM, nous envisageons des expériences avec des ensembles de données similaires (par défis et nature des données), comme les collections de données multilingues et les données médicales.

Abstract

Advanced Process Control (APC) is a research direction in the Semiconductor Manufacturing (SM) industry engaged in developing automotive process diagnostic and product management solutions to preserve high end-of-line yield and reduce the risk of equipment failure.

In this thesis, we focus on investigating Machine Learning (ML) based approaches to developing a unified framework for process analysis and modeling from highly diversified multi-view (that comes from different sources) SM data. Overall, multi-view features give a more comprehensive description of a phenomenon, and a well-designed multi-view learning strategy has better generalization ability than single-view learning, which justifies an appropriate research effort.

One of the leading open challenges in the APC field in SM is the ability to leverage the wealth of information in order to fully characterize the process and determine the value of new measurements. Accordingly, we start with the analysis of existing data treatment techniques and their limitations. Then, we focus on proposing a strategy that consists of data cleaning, features extraction, and features selection steps to deal with data imperfections usually expected in the field, like noise, irregular sampling steps in sensory time series data, and incomplete records, all due to natural corruption-error rate of recording tools.

Next, this thesis intends to expand the scope of traditional process modeling in SM by cross-process analysis. Product manufacturing is a sequential procedure of applying ordered processes to deposit new layers of features; then, one can use the precedent history to learn its impact on the current modeling target of interest. Accordingly, we propose a methodology that benefits not only from different types of measurements but from dependencies between different process steps to make processes more predictable and productive.

Moreover, we study the problem of missing data, mainly when one of the views is entirely missing, which is another open challenge in the field. Some studies tackle this problem by assuming the existence of view generation functions to approximately complete the absent views. However, these functions generally require an external resource to be set, and their quality directly impacts the performance of the final predictive model learned over the completed training set. Instead, in this work, we propose to address this problem by jointly learning the missing views and the multi-view target estimator using an adversarial learning approach inspired by the ability of Generative Adversarial Networks Generative Adversarial Networks (GANs) to seize the underlying distribution of the data and create new samples.

Finally, for all the hypotheses introduced above in this work, we consider the APC tasks like Virtual Metrology (VM) and Predictive Maintenance (PdM) to conduct experiments

using the real data collections provided by leading in Europe Semiconductor Manufacturing fabrication facilities that we collaborated with, within the scope of the Metrology Advances for Digitized Electronic Components and Systems (ECS) Industry 4.0 (MADEin4) Project. Moreover, since the problem of missing data in multi-view collections is widespread in different data sets beyond the SM industry, we consider experiments with similar data sets (by challenges and data nature), like multi-lingual data collections and real world Electronic Health Record (EHR) data.

Contents

Acronyms

R^2 Coefficient of Determination.

AE Autoencoder.

ANN Artificial Neural Network.

APC Advanced Process Control.

CCA Canonical Correlation Analysis.

ccGAN clinical conditional Generative Adversarial Network.

CDO Chemical Oxide Deposition.

cGAN conditional Generative Adversarial Network.

CNN Convolutional Neural Network.

CoGAN Coupled Generative Adversarial Network.

CuECD Copper Electroplating Deposition.

DAE Denoising Autoencoder.

DNN Deep Neural Network.

DT Decision Tree.

EHR Electronic Health Record.

ESD Equipment Sensory Data.

FFNN Feed Forward Neural Network.

FN False Negative.

FP False Positive.

GAIN Generative Adversarial Imputation Net.

GAN Generative Adversarial Network.

GBDT Gradient Boosted Decision Trees.

GMM Gaussian Mixture Models.

IC Integrated Circuit.

KNN k-Nearest Neighbors.

LR Linear Regression.

LSTM Long Short-Term Memory Network.

MADEin4 Metrology Advances for Digitized Electronic Components and Systems (ECS) Industry 4.0.

MAPE Mean Absolute Percentage Error.

MICE Multivariate Imputation by Chained Equations.

MissF MissForest.

MKL Multiple Kernel Learning.

ML Machine Learning.

MLR Multiple Linear Regression.

MT Machine Translation.

PdM Predictive Maintenance.

PvM Preventive Maintenance.

RBFN Radial Basis Function Network.

RF Random Forest.

RMSE Root Mean Square Error.

RUL Remaining Useful Life.

SLR Simple Linear Regression.

SM Semiconductor Manufacturing.

SVM Support Vector Machines.

SVR Support Vector Regression.

VAE Variational Autoencoder.

VCM Virtual Cross Metrology.

VM Virtual Metrology.

W2W Wafer-to-Wafer.

XGB eXtreme Gradient Boosting.

Chapter 1

Introduction

1.1 Research Context

A Semiconductor Manufacturing (SM) is an industry that is engaged in designing and fabrication of Integrated Circuits (ICs), or so-called chips. They are sets of interconnected microelectronic components, such as resistors, transistors, capacitors, and inductors, made of layers deposited on tiny (10 millimeters scale on average) but complex single units. Their manufacturing starts with an empty wafer sawed out of an ingot of pure crystalline silicon or other semiconductor material, designed in the form of a 300 millimeters disc. Then, a rectangular pattern is printed on the wafer, where each small block, called die, serves for growing there the IC, and it takes from 90 up to 150 days to assemble normally 30 or more layers of nanoscale features toward the final IC. Figure 1.1 describes the entire process flow of enabling silicon wafers into functional chips in a manufacturing facility - fab. Particularly, sets of chemical and physical process operations like polishing, material deposition, modification, metallization, lithography, etching, and more are performed in a chain to build each new layer of microelectronic features upon the wafer. Then, depending on the design requirements for the number of different feature layers, such operations are repeatedly performed several times to gradually convert raw materials into finished electronic products. Once fabricated, the wafer undergoes electrical testing, or the so-called wafer acceptance test (WAT), to discard dies on the wafer with a lack of requested functionality. The fraction of accepted (successfully produced) chips is called yield. Finally, qualified products are packaged and shipped in the electronic systems for further use.

Nowadays, SM companies tend to move to high-volume production environments as they follow an increasing demand for electronic components in the modern world. It implies the development and utilization of specialized material technologies and modern equipment that allow the production of denser wafers of more complex designs for ICs. Then, with the introduction of new, often marginal and difficult-to-control processes, into advanced manufacturing, reaching a sufficiently high and competitive productivity level has become, and will continue to be, a serious challenge. The value of wafers is increasing; therefore, the impact of equipment failures resulting in tool downtime and loss of wafers is a major concern.

Accordingly, mastering productivity within the SM industry has two main research

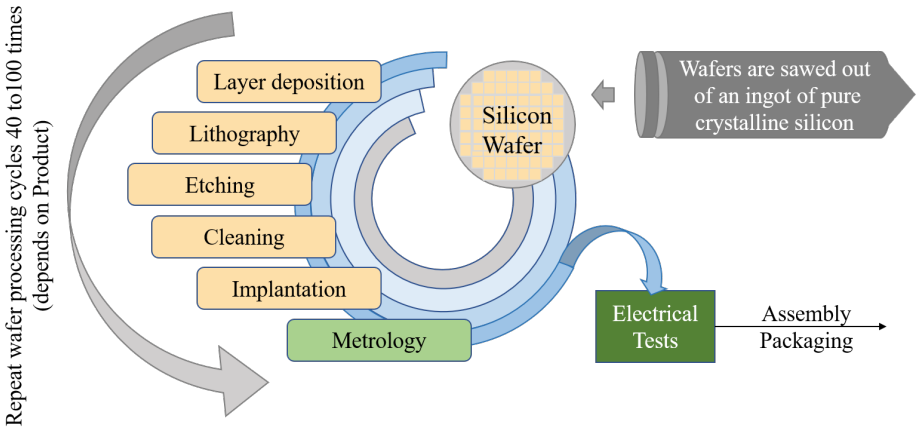


Figure 1.1: Schematic representation of the SM ecosystem.

directions to develop:

- strategies for *equipment maintenance* to reduce the risk of equipment failure;
- solutions for *product monitoring* to preserve high end-of-line yield.

1.1.1 Equipment Maintenance

Equipment maintenance is crucial to guarantee the proper functioning of performing tools. Any unplanned downtime of manufacturing machinery at any stage of the fab ecosystem is crucial for the business, as it downgrades the production capacity and demands additional costs to repair or sometimes even replace ruptured components. Therefore, systematic (pre-scheduled) equipment inspection - Preventive Maintenance (PvM), was introduced to prevent unexpected outages. According to PvM, manufacturing tools are being controlled by relying on conventional mathematical models and programming methods, so that maintenance is scheduled when predefined statistically selected criteria is reached (like duration of machine's running time hits a threshold of the average equipment operational time without any abnormalities). However, an obvious limitations of PvM for the process control is the possibility of conducting unnecessary maintenance and an inability to detect and prevent failures that may happen before the planned maintenance time.

1.1.2 Product Monitoring

Regarding product monitoring, a domain of metrology analysis could be considered as the 'eyes and ears' in the SM fabs because its requirement is to support all processes steps toward the final product. In particular, at each of the production stages, an excellence of corresponding process equipment performance can be monitored by measuring critical characteristics of the deposited new layer on processed wafers, such as thickness, stress, concentration, critical dimension, and more - altogether called metrology. Such analysis assures the quality of produced devices leading to high end-of-line yield and stability of process equipment. However, the large volume of multi-stage manufacturing systems and

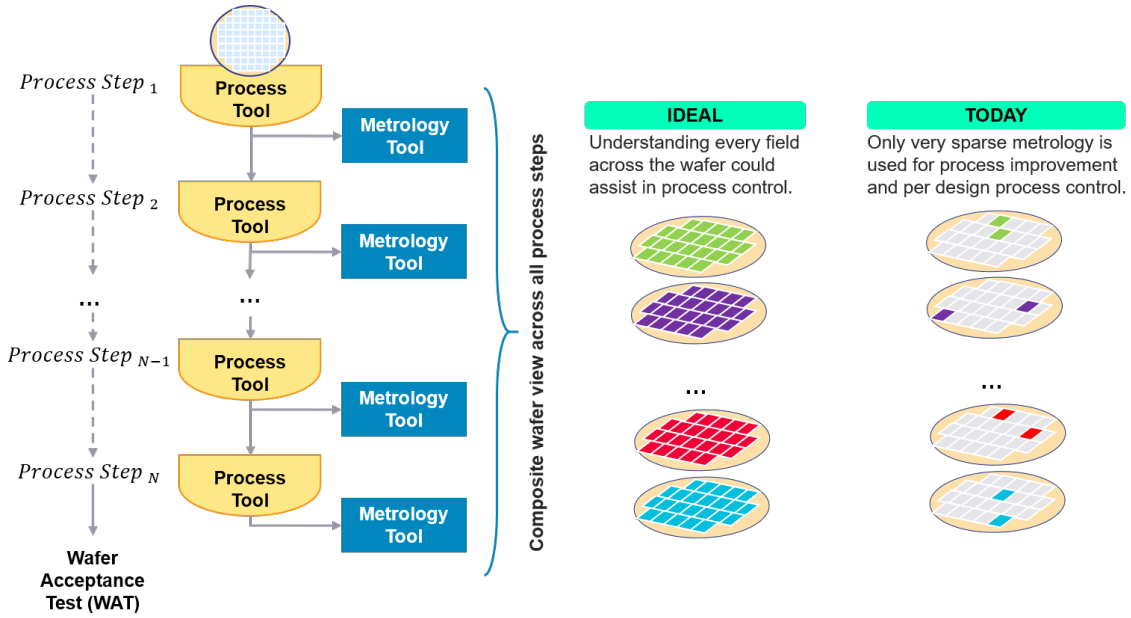


Figure 1.2: Schematic representation of the full chain of processes applied to one wafer to fabricate ICs on it. Then, in order to perform a high-quality process control at any stage, full coverage of all of the different metrology measurements is needed. However, in practice, collecting metrology for each wafer causes a high cost of production and significantly increases the fabrication cycle time. Therefore, a common method of monitoring consists of using only a few periodically sampled wafers and measuring there only a few dies.

nanometric 2D and 3D scales of taken measurements makes metrology monitoring as one of the challenging forms, time-consuming and expensive. Therefore, in practice, only very sparse metrology is used for process control, described in Figure 1.2. It means that for any given electronic component and system technology, there is a significant trade-off between its productivity to its metrology precision and accuracy. For example, gaining in precision and accuracy reduce the measurement cycle time (productivity) significantly, and cause a yield risk to the final product due to low sampling.

1.2 Motivation

On the other hand, Advanced Process Control (APC), in its turn, is a likely productivity booster in manufacturing sectors that use Machine Learning (ML) model as a base approach to exploit the information already present in the SM process system, like physical sensors measurements, tool settings or design characteristics, in order to infer the value of a costly or unmeasurable variable, like metrology, which is important for the decision making in process control or for characterizing the production quality. Usually, this goal is achieved by means of supervised learning methods where a ML model is created by leveraging labeled data where both the input and the output have been physically measured from past process runs. In this way, APC solutions facilitate a replacement of conventional PvM with Predictive Maintenance (PdM) schedules activities that are based on collected data from sensors

and analysis algorithms; and improve the total metrology coverage by providing values at other sample locations in addition to those currently performed - Virtual Metrology (VM) technology. Together PdM and VM solutions help limit the number of human interventions, minimize equipment downtime, lower the number of defects in produced items, and improve overall reliability.

For the past three years, the Metrology Advances for Digitized Electronic Components and Systems (ECS) Industry 4.0 (MADEin4) Project has developed next-generation processes and metrology tools, ML methods and applications in support of Industry 4.0, also known as smart manufacturing for the semiconductor fabs. Within the MADEin4 Project, we collaborated with the teams of European leading SM companies working on key application areas such as the automotive industry to

1. investigate and demonstrate the shortcomings of the real industrial data collections from the SM fabs for PdM and VM tasks;
2. analyse different ML methods commonly used in PdM and VM tasks;
3. propose and deliver innovative solutions for APC framework consisting of the software architectures for processes like data gathering, data pre-processing, and data analysis in SM industry.

1.2.1 Problem Formulation

1.2.1.1 Predictive Maintenance (PdM)

PdM, also known as condition-based maintenance (CBM) [29], that typically involves condition monitoring, anomaly detection, fault prognosis, and maintenance plans, aims to predict when the equipment is likely to fail and decide which maintenance activity should be performed such that a good trade-off between maintenance frequency and cost can be achieved. Particularly, PdM is engaged in developing methodologies for identification if a system status is considered anomalous or faulty through ML models as a function of process-related data. Accordingly, at considered time t the estimated system status $\hat{s}(t)$ could be expressed as

$$\hat{s}(t) = f(p_1(t), p_2(t), ..., p_m(t), u(t)) \tag{1.1}$$

where $p_1(t), p_2(t), ..., p_m(t)$ refer to measurements of m different sensor parameters of considered system at time t, and $u(t)$ consists of any additionally available information about operating equipment, like a recipe - set of tool instructions which specify how a processing step is to be performed.

SM industry has reported that with the introduction of advanced anomaly detection, a fab could have 25% reduction in time to yield maturity, 10% increase in manufacturing capacity, and 35% decrease in a number of quality problems [17] that justifies an appropriate research effort.

1.2.1.2 Virtual Metrology (VM)

Every machine in a fab is equipped with sensors for regular automatic measurements of process conditions inside the operating chamber. This data, considered as process variables,

is automatically saved and already used for manually driven conventional programming methods of APC. However, additionally to that, recent state-of-the-art metrology domain developments propose to apply ML solutions with the data collected form the equipment sensors for metrology values estimation - VM. This direction of research and development, also known as a soft sensor, is engaged in predictive diagnostics of the process and tool performances. Accordingly, VM is engaged in developing robust methods for metrology output in the function of process variables and other information available for the process and/or the product, so that the estimated target metrology measurement at the (x, y) location of i-th wafer w_i on the tool, $\hat{m}(w_i, x, y)$, is given by

$$\hat{m}(w_i, x, y) = f(p_1(w_i, x, y), p_2(w_i, x, y), ..., p_m(w_i, x, y), u(w_i, x, y)) \tag{1.2}$$

where $p_1(w_i, x, y), p_2(w_i, x, y), ..., p_m(w_i, x, y)$ are the parameter records taken from m in total chamber sensors during the processing of wafer k, and $u(w_i, x, y)$ that refers to any other available auxiliary information, like recipe or design - set of product specifications.

Some studies state that the implementation of VM in a fab is estimated to increase the production volume output by nearly 10% [21] that justifies an appropriate research effort.

1.2.2 Challenges

Although, ML models are being widely investigated in the semiconductor manufacturing field for VM and PdM tasks, a few of them are actually being deployed in the fab, and not to substitute existing conventional methods for equipment maintenance and product monitoring, but to augment them. This is due to the following challenges that are present in the field.

Challenge 1. Abundant multi-view process characterization data treatment.

Manufacturing machinery is equipped with recording tools and sensors measuring process state-related information, which is ready to be used without any delays. This data comes from various sources and is of different types, like Equipment Sensory Data (ESD) that consists of recordings of sets of parameters from process sensors, recipe information, and other numerical and categorical process/equipment characteristics; design requirements; defects inspection maps; past maintenance diagnostic results; history of equipment failures, and more. Then, in order to assure the stability and effectiveness of the developed ML solutions based on such *highly diversified data*, it is necessary to perform preliminary data transformation of the raw records collections.

For example, ESD forms the core of industrial SM data sets. In particular, the data is given by the time sequences of several sensors that consist of hundreds of recorded values. This means that one wafer ESD consists of thousands of features on average. Accordingly, it is a problem where the features space dimension is of the same order as the number of samples, since the number of labeled observations (wafers) in the SM data collections for supervised ML strategies reaches a few thousand maximum due to high cost of extracting the labels. Therefore, measures must be taken to face the *high dimensionality* of the problem, both for the accuracy of the solutions and to avoid oversmoothing.

Then, *missing data* and *outliers* are to be expected due to natural tool variability and since recording tools of different sensors may collect data unevenly and with a fluctuating

time step that usually may vary in a tolerance range of few seconds/minutes. Besides, some records are expensive and time-consuming to measure due to their nanometric scale and precision, and therefore only a subset of observations shall be subject to corresponding analysis, while the rest is left without the data being provided.

Outliers, in turn, are observations significantly distinct from a given population of records and are very common in different data collections beyond semiconductor industrial sets. They can be efficiently detected by statistical tests, and usually, analysis shows that outliers are due to the natural corruption-error rate of recording tools, so that can be removed for further problem research. However, it happens that such observations actually indicate anomalies in tool functioning. Then, in this case, since productivity is the primary concern, such records are required for further investigation. And therefore, it is necessary to provide a reliable mechanism for classifying abnormal observations for data-cleaning purposes without losing any critical information.

Challenge 2. Generalization guarantees for VM and PdM predictive models.

The constant change and introduction of new products and recipes to production require predictive models to be retrained automatically as new data comes in to meet generalization guarantees. Besides, some process hardware exhibits a drift that implies a distribution shift specifically of the sensory data over time. Therefore, it is essential to define *maintenance/re-calibration conditions* with deployed ML framework for VM and PdM tasks in the SM fab in order to preserve their *reliability*.

Challenge 3. Interpretability.

Analyzing the existing process monitoring schemes, the prediction accuracy of the process status is usually the primary focus, while the explanation (diagnosis) of a detected fault is relegated to a secondary role. Nevertheless, model interpretability is considered to be an important issue because engineers who actually operate the semiconductor tools prefer a model that can be easily understood and displays the underlying causal interactions of a process system in an easily interpretable graphical form.

1.3 Thesis Structure

In this thesis, we showcase a practical ML methodology for PdM and VM problem that combines process, metrology, and design information within a single framework to learn about the process and design contributions affecting manufacturing, with the ultimate objective of the research to have a sufficiently complete virtual representation of the process and the measurements that define if the process is operating within the pre-defined budgets, and provide the infrastructure to enable process analysis, defect analysis, process optimization, and process control. Additionally, we focus on investigating the value of missing information in the SM data collections to affect the accuracy of the developed predictive strategies and propose the methodologies to address the challenge.

The rest of the thesis consists of the following.

- In Chapter 2, we provide a review of different existing ML approaches that are commonly applied for building VM and PdM predictive applications, which includes

both: methods for data pre-processing and methods for target modeling. Moreover, in this Chapter, we review different solutions in literature for the multi-view ML applications and how they are affected by the missing data. In this respect, we provide an investigation on the Generative Adversarial Networks (GANs) that allows reconstructing the missing view from the available ones by solving the domain transfer problem.

- Chapter 3 describes activities within the MADEin4 Project related to the development of the PdM framework for modeling process drift from its initial state that is observed after the latest PvM event in order to detect deviations from normal conditions. The proposed approach consists of two parts: features extraction and modeling, which both leverage several state-of-the-art statistical and ML methods integrated into a rather complex framework. First, we operated the raw parameters data with a windowing strategy, Gaussian Mixture Models (GMM) split into signal modes and centering with respect to PvM events to remove all the possible sources of systematic variations retaining only random variations and to have a compact and uniform representative as much as possible information on the process state in order to generate better quality input for the modeling part. Then, the core approach for a predictive diagnostic of the tool in our work is based on the Gradient Boosted Decision Trees (GBDT) that is an efficient method for regression and classification tasks in ML from the family of ensemble algorithms.

- Chapter 4 is based also on the MADEin4 Project case studies, where we present a fusion of electronic design, process, and metrology data ML-based framework for improved process optimization and control with VM modules for individual process applications. Moreover, in this Chapter, we propose to expand the scope of traditional process modeling in the SM by cross-process analysis, called Virtual Cross Metrology (VCM).

- In Chapter 5 we discuss the problem of missing data in the multi-view ML frameworks when the view may be missing completely. We introduce a conditional GAN model with two generators and a common discriminator for multi-view learning problems where observations have two views, but one of them may be missing for some of the training samples. Experimentally, we show that the approach that jointly learns the missing data imputation and target estimation leads to better performance (based on the target accuracy estimation) compared to the methodologies that consider the two tasks separately.

- In Chapter 6 we discuss the problem of partially missing data in the multi-view ML frameworks. Particularly, we study the problem with Electronic Health Record (EHR) data and we present a data imputation technique based on a clinical conditional GAN (ccGAN) capable of imputing missing values of observed characteristics conditioned by fully-available characteristics values to be then employed for predicting the probable diabetes complication.

Personal References

[Ana21] Anastasiia Doinychko, Umberto Amato, Stanislau Raitsyn, Stefania Perna, Franco Blundo, Caterina Genua, Daniele Vinciguerra, Antonino La Magna, Andres Torres, Alex Rosenbaum, Massih-Reza Amini, and Patrizia Vasquez. Virtual metrology to eliminate test wafers measurements on copper electroplating deposition. In *IEEE International Conference on Automation Science and Engineering (CASE) Lyon, France*, 2021.

[Ana22a] Anastasiia Doinychko, Andres Torres, Ivan Kissiov, Melody Tao and Sanghyun Choi. A fusion of electronic design, process and metrology in semiconductor manufacturing for improved process optimization and control. In *International Society for Business ans Industrial Statistics (ISBIS) Naples, Italy*, 2022.

[Ana22b] Anastasiia Doinychko, Andres Torres, Ivan Kissiov, Melody Tao and Sanghyun Choi. Improved semiconductor process characterization using virtual cross metrology. In *Advanced Process Control Smart Manufacturing (APCSM) TX, USA*, 2022.

[DA20] Anastasiia Doinychko and Massih-Reza Amini. Biconditional generative adversarial networks for multiview learning with missing views. In *42^{nd} European Conference on IR Research, (ECIR)*, pages 807–820, 2020.

Under Review

[AAF+22] Umberto Amato, Anestis Antoniadis, Italia De Feis, Anastasiia Doinychko, Irene Gijbels, Antonino La Magna, Daniele Pagano, Francesco Piccinini, Easter Selvan Suviseshamutu, Carlo Severgnini, Andres Torres, and Patrizia Vasquez. Prediction of yield in semiconductor production from defects detected by sem on the wafers. *Journal of Applied Stochastic Models in Business and Industry*, 2022.

[ADT+22] Thomas J. Ashby, Anastasiia Doinychko, Andres Torres, Daniele Pagano, Wilfried Verachtert, and Roel Wuyts. Paml for virtual metrology. *IEEE Advancing Semiconductor Manufacturing Excellence (ASMC)*, 2022.

[BDaEFA22] Michele Bernardini, Anastasiia Doinychko, Luca Romeo ands Emanuele Frontoni, and Massih-Reza Amini. A novel missing data imputation approach based on conditional generative adversarial networks applied to a real challenging ehr dataset. *Journal of Biomedical and Health Informatics*, 2022.

Chapter 2

Related background

In this Chapter we provide a review of different existing ML approaches applied for building VM and PdM predictive applications. Overall, both frameworks consist of two main parts: data treatment and modeling, which are equally important for building accurate estimators. Accordingly, in Section 2.1 we, first, discuss data pre-processing methodologies and features extraction techniques for transformation of raw semiconductor collections of records into a proper data format for building an efficient ML model from. Next, in Section 2.2 we provide a survey on different families of ML algorithms for VM and PdM problems solving.

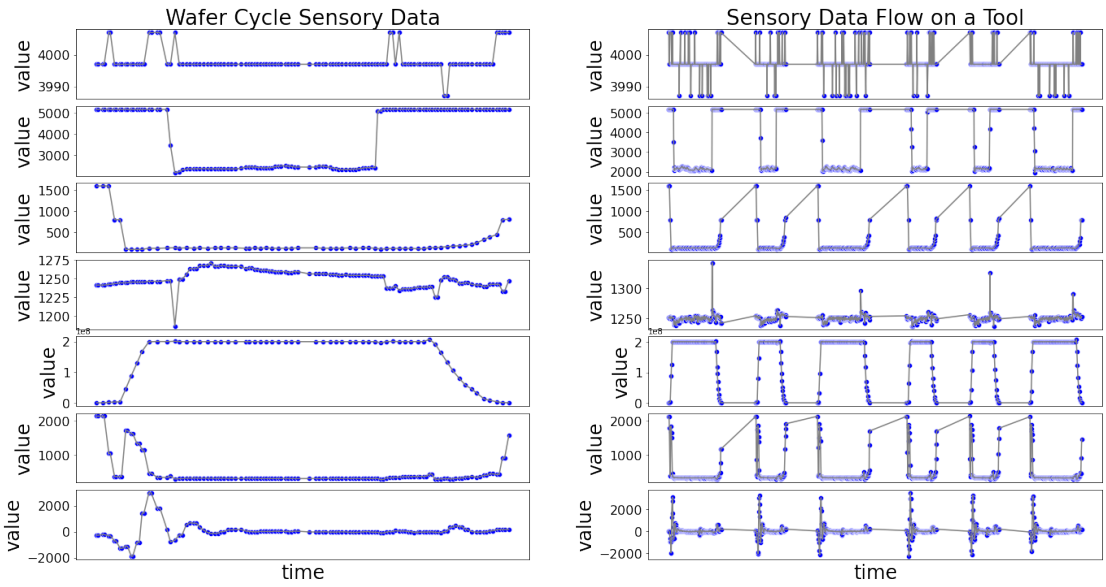
Finally, in Section 2.3 we describe the state-of-the-art methods to deal with one of the main problems in SM data collections as well as in many industrial data sets in general - missing data. In particular, we consider the case of multi-view ML tasks when during training some samples may have one of the views missing. Accordingly, we provide a review of the multi-view learning strategies in ML and approaches that usually are employed to overcome data incompleteness.

2.1 Process Characterization in Semiconductor Manufacturing Data Collections

2.1.1 Equipment Sensory Data (ESD)

One of the most important factors that lead to a more sustainable VM or PdM strategy in a production environment is the level of data pre-processing required. As anticipated in the Introduction, the SM data for VM and PdM tasks is an abundant multi-view collection. The core of it is Equipment Sensory Data (ESD) collected from sensors mounted on the manufacturing equipment, and it is one of the challenges to make effective use of ESD due to its irregular sampling, high dimensionality, presence of outliers, missing data and impermanent length.

In a fab, a recording of ESD is triggered by a presence of a wafer in a processing tool, then it lasts until all operations on the wafer are finished, which usually takes a few minutes and generally depends on a recipe. As a result, one parameters data sample $\mathbf{p}_i$ - one wafer process cycle ESD - is generated for a wafer w_i (while it was treated in the processing equipment), which is a collection of sensory parameter values as a function of time. If P is



(a) (b)

Figure 2.1: (a) Example of a one wafer cycle ESD that consists of recordings of 7 different sensory parameters. (b) A fragment of several ESD cycles that consist of recordings of 7 different sensory parameters collected from 6 subsequently processed wafers on the tool. (Both figures were generated from the real fab data analyzed in the MADEin4 Project.)

a total number of sensory parameters, then $\mathbf{p}_i$ is defined by Equation 2.1.

$$\mathbf{p}_i(w_i) = \{\mathbf{p}^s(w_i)\}_{s=1}^{P} \tag{2.1}$$

Then, each $\mathbf{p}^s(w_i)$ is a time series:

$$\mathbf{p}^s(w_i) = \{p^s(t_T(w_i))\}_{T=0}^{N(s,w_i)},\ t_0(w_i) \leq t_T(w_i) \leq t_0(w_i) + \Delta t(w_i),$$

where $t_0(w_i)$ indicates the time when the process started for the wafer w_i and $\Delta t(w_i)$ denotes its duration. It is to be stressed that the time sequences $\mathbf{p}^s$ are not equispaced in time

$$t_{T+1}(w_i) - t_T(w_i) \neq const\ ,\ \forall T \in \{1, ..., N(s, w_i)\}\ \forall s \in \{1, ..., P\} \tag{2.2}$$

and may not be aligned, because the recording system collects sensory values with a fluctuating time step that usually may vary in a tolerance range of a few seconds. Figure 2.1a shows the example of one wafer ESD from the data collection of the MADEin4 Project analyzed in this thesis, and it is clear that the sampling frequency varies depending on the parameter.

After one wafer is processed, the recording system stays in "waiting" regime until the next wafer comes into the tool. The same tool can process wafers of several designs, and therefore it is common that different recipes are switched on a tool between processing consecutive wafers of different designs. As a result, duration $(t_{N(s,w_i)}(w_i) - t_0(w_i))$, and as a consequence length $|\mathbf{p}^s(w_i)|$, of the ESD samples is different for every wafer w_i due

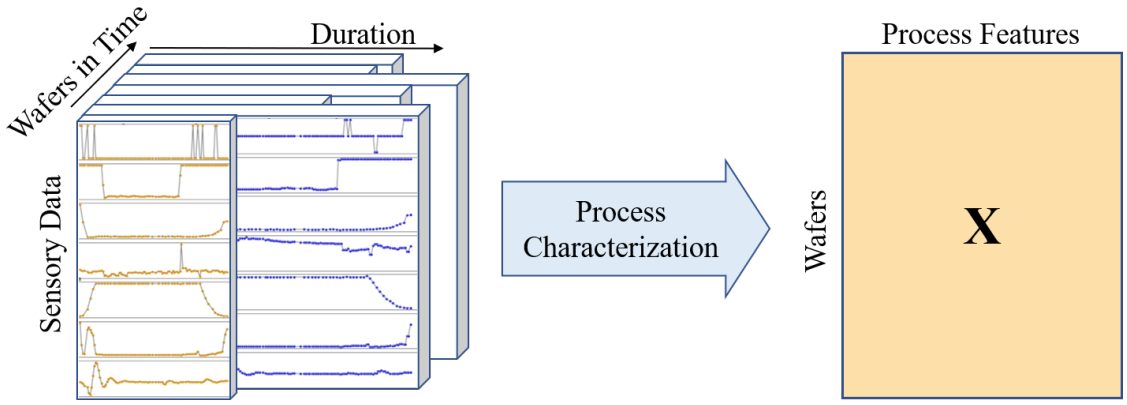


Figure 2.2: Preliminary operation, called feature extraction or in case of ESD - process characterization, where a set of informative values is extracted from the raw data, and then casted into a design matrix $\mathbf{X}$.

to the presence of different recipes and natural variability of the sampling frequency of the recording tool (Figure 2.1b).

Traditional ML techniques that are usually employed in this context are not suitable to deal with these highly inconsistent input data, while high dimensionality is the additional concern. In fact, the average number of data (length) for each ESD sequence sample (that corresponds to one wafer and one type of sensor parameter) is around hundreds of records. Then if tens of different parameters are to be collected, a process characterization set for one wafer contains around thousands of data in the average. In practice, there exists a problem where the number of samples is of the same order as the number of explanatory variables (Equation 2.3).

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{s=1}^{P}|\mathbf{p}^s(w_i)| \geq N \tag{2.3}$$

Therefore measures have to be taken to face the high dimensionality of the problem, both for the accuracy of the solutions and to avoid oversmoothing.

2.1.2 ESD Feature Extraction Methodologies

One of the most popular methods for reducing the number of variates is to deal with the features extracted from the process signals instead of the original ones, casted into a matrix $\mathbf{X}$ (Figure 2.2). Let g^s denote a transformation applied to the sensory parameter s, $s \in \{1, ..., P\}$, then $\mathbf{X}$ is defined as

$$\mathbf{X} = \left\{ \left(g^1(\mathbf{p}^1(w_i)), g^2(\mathbf{p}^2(w_i)), ..., g^P(\mathbf{p}^P(w_i)) \right) \right\}_{i=1}^{N} \tag{2.4}$$

and the outcome of $g^s(\mathbf{p}^s(w_i))$ is constant across all the wafers w_i, $i \in \{1, ..., N\}$. Then, if $\mathbf{x}^s(w_i) = g^s(\mathbf{p}^s(w_i))$ and $|g^s(\mathbf{p}^s(w_i))| = d_s$, the $\mathbf{X}$ can be express in the following form

$$\mathbf{X} = \left\{ \left(\mathbf{x}^1(w_i), \mathbf{x}^2(w_i), ..., \mathbf{x}^P(w_i) \right) \right\}_{i=1}^{N}, \ \mathbf{x}^s \in \mathbb{R}^{d_s}. \tag{2.5}$$

The most traditional method of features extraction consists of computation of relatively basic statistical and descriptive characteristics computed on a set of values of each ESD sequence sample of each sensor parameter, such as mean, variance, skewness, kurtosis, peak-to-peak values, step durations and more. Particularly, features described in Table 2.1 are commonly used for PdM and VM models development, and are intended to catch the main characteristics of the signal, also attempting to get basic local information on the sequence. As a result, feature-based PdM and VM strategies naturally address common challenges like high dimensionality as well as unequal signal lengths and/or unsynchronized wafers process trajectories. Moreover, in [65] study, the authors argue that the feature-based approaches can better capture process characteristics and dynamic behaviors.

Nevertheless, more advanced features are considered as well, which aim at picking more detailed information on the signals. For example, a well-known in the literature, the wavelet transform gives a very accurate representation of the signal in time and frequency, and therefore wavelet theory has been widely used in fault detection [77] and process monitoring [16] applications. Roughly speaking, it is able to give the local role of the frequency composing a signal by means of a linear combination of some special basis functions (wavelets) that are obtained from a unique function by dilation and translation. Wavelet representation offers excellent theoretical properties in terms of approximation of a function and decorrelation of the coefficients of the representation. Efficient discrete wavelet transform is basically defined for dyadic (length power of 2) and equispaced time sequences, which is may not be the case for the ESD collections in some applications, like ones that we worked with during the MADEin4 Project. Therefore a special approach was developed in [1, 2] to handle such cases of non-equispaced sequences and of generic length.

Next, the Fourier transform shares many of the properties of the Wavelet transform, in that it gives a representation of a signal in terms of a basis of trigonometric functions rather than wavelet ones. Such basis functions represent frequencies that compose the signal. Fourier transform has worse theoretical properties than Wavelets as far as the degree of approximation and decorrelation of coefficients of the basis trigonometric functions are concerned; in addition, it is not a local representation of the signal, in that each coefficient represents a frequency all over the domain of the function. However, it can be effective for functions showing oscillatory behavior. From the computational point of view, also (discrete) Fourier transform is basically defined for dyadic and equispaced time sequences. To work with nonequispaced and nondyadic sequences an approach can be pursued similar to the Wavelet transform proposed in [1, 2].

Recently, more sophisticated automatic feature extraction methods, like Autoencoders (AE), has been proposed for SM process characterization tasks [48, 47, 78]. Generally, the AE is a Artificial Neural Network (ANN) that is trained to reconstruct its input. In the particular case of time-series based task, Convolutional Neural Network (CNN) [78, 34] and Long Short-Term Memory Network (LSTM) recurrent neural network are employed in order to effectively learn from the sequential type of input that usually require some preliminary operations, like resampling or moving average, in order to make them equispaced. Then, the hidden layers of the network usually perform dimensionality reduction on the input, learning relevant features that allow a good reconstruction. Moreover, deep AEs exploit multiple non-linear representational layers that learn complex hierarchical features from the data with high informative content [48, 47]. However, the use of deep models requires

Feature Name	Significance
AverageValue	Average of the parameter's values along the entire time sequence.
AverageLeft	Average of the parameter's values along the first half of the time sequence.
AverageMiddle	Average of the parameter's values along the central 50% part of the time sequence (discarding the 25% left and right parts)
AverageRight	Average of the parameter's values along the second half of the time sequence.
AverageDelta	Difference *AverageRight – AverageLeft*.
FirstValue	First recorded value in the parameter's time sequence.
LastValue	Last recorded value in the parameter's time sequence.
Kurt	Kurtosis of the parameter's values along the entire time sequence.
Skew	Skewness of the parameter's values along the entire time sequence.
MedianValue	Median of the parameter's values along the entire time sequence.
SD	Standard deviation of the parameter values along the entire time sequence.
SDLeft	Standard deviation of the parameter values along the first half of the time sequence.
SDMiddle	Standard deviation of the parameter values along the central 50% part of the time sequence (discarding the 25% left and right parts).
SDRight	Standard deviation of the parameter values along the second half of the time sequence.
ValueMin	Minimum of the parameter values along the entire time sequence.
ValueMax	Maximum of the parameter values along the entire time sequence.
Time1	First available time of the parameter's time sequence.
Time2	Last available time of the parameter's time sequence.
Duration	Difference *Time2 – Time1*.
Length/Size	Total number of values collected in the entire time sequence.
TimeMin	The time when the minimum value of the time sequence recorded.
TimeMax	The time when the minimum value of the time sequence recorded.
Area	Total area under the curve of the parameter's time sequence.

Table 2.1: Description of the commonly used statistical and descriptive features extracted from SM sensory data for process characterization.

big data collection during training which is not all the time the case with the SM data sets. Moreover, another limitation of the AE feature-extraction approaches is the difficulty in extracting process knowledge from the new feature space once it has been transformed, so that we are compromising on interpretability.

2.1.3 Features Selection Methodologies

Some of the extracted features (especially statistical ones) can measure similar quantities, therefore, can show high correlations. Then, there is a high chance that the performance of a predictive model can be impacted by a problem called multicollinearity, which may lead to possibly unstable predictive models. To overcome the obstacle, a lot of feature selection mechanisms are proposed in the literature. For example, in [53], authors compared state-of-the-art model-building techniques such as Forward Selection Regression (FSR), Ridge regression, LASSO, and Forward Selection Ridge Regression (FCRR); and validated them on a benchmark semiconductor plasma etch dataset in highly correlated input spaces, showing that the FSR provides the best result.

A method of fitting multiple regression models known as stepwise regression offers a way to choose the optimal subset of explanatory variables. In particular, one predictor is added at a time until no more increase in the selected accuracy score is observed. The estimate of coefficients of the predictors and of the accuracy score has to be performed on separate data sets: training and testing ones respectively, because otherwise the accuracy always decreases with the number of explanatory variables. A semi-exhaustive trial method is used to include more predictors; at each stage, all variables that have not previously been incorporated into the model at a previous phase are tested, and the variable (if any) with the highest accuracy score is chosen — a process known as forward selection. In contrast, there is another variation of the stepwise regression by backward elimination, when the procedure starts with fitting the regression model with a full set of predictors and then iteratively one variable is removed at a time.

Other methods use regularization that relies on functional arguments to select variables. The rationale comes that for the observation in high-dimensional settings, the solution can be controlled by introducing ancillary constraints under the form of a regularizing function and a corresponding regularization coefficient. In mathematical terms, the quadratic optimization problem that is at the basis of regression is replaced by the following regularized problem

$$\min_{w} \|\mathbf{y} - \mathbf{w}\mathbf{X}\|_2^2 + \lambda p(w), \tag{2.6}$$

with p and λ being, respectively, the regularization function and the regularization coefficient. The prototype of such regression methods is LASSO, where $p(w) = \|w\|_1$. This regularization term yields sparse solutions, in the sense that some (possibly most) coefficients are set to 0, and therefore corresponding predictors do not enter the model and are not selected. Some generalizations of LASSO have been proposed aimed at fixing the bias inherent in LASSO and at improving accuracy, through different choices of the regularization function and even replacing the L_2 norm with different loss functions.

In all methods a second regularization term may also be included as the L_2 norm with

its regularization coefficient (elastic net):

$$\min_{w} \|\mathbf{y} - \mathbf{w}\mathbf{X}\|_2^2 + \lambda p(w) + \alpha \|w\|_2^2. \tag{2.7}$$

which fixes the drawbacks of regularization methods with a high number of predictors and highly correlated predictors. Both regularization coefficients λ and α are estimated by cross-validation.

2.2 Modelling

Maintenance issues can be completely different in nature, as well as metrology predictive tasks may have different specifications depending on the type of target. Therefore, the predictive information to be fed to the PdM or VM module has, in general, to be tailored to the particular problem at hand. This observation justifies the presence in the overview papers of both PdM [57] and VM [72, 25] of many different approaches that are discussed further in this Section.

2.2.1 Predictive Maintenance (PdM)

The PdM is engaged in developing methodologies for identification if a system status is considered anomalous or faulty through ML models as a function of process-related data. However, it is very rare that the PdM task is considered as a binary classification problem, since this scenario requires a sufficient number of examples in both categories ("faulty" and "normal" observations) in the training data. Generally, error events are very rare to be observed since a lot of efforts in a SM fab are directed to prevent breakdowns, which makes corresponding data sets to be hugely unbalanced or skewed. Nevertheless, in [14, 64] authors suggest choosing larger values for the failure horizon, so that instead of only labeling the last iteration before the error event as "faulty", they label the last n iterations (wafers that equipment produced before it went off). Then overall, the such methodology does not impose any restrictions on the choice of a classification algorithm. In literature, Support Vector Machines (SVM), k-Nearest Neighbors (KNN) or ANN were demonstrated to provide competitive results [64, 52, 14]. However, Decision Tree (DT) ensemble models may be preferable in case of major class imbalance, since this ensemble algorithm allows to form subsets with majority class down-sampling to train trees. Moreover, in [64] authors propose to repeat the procedure for k different values of the horizon n to build k different classifiers respectively, each one facing a different classification problem and therefore providing different performance outcomes.

Next, regression-based formulations of PdM are more frequent in practice as described in [57] and generally arise when predicting a Remaining Useful Life (RUL), time-to-failure (TTF), or health indicator (HI) using commonly applied traditional regression ML models, like SVMs, ANNs, DTs and KNNs. However, training to predict RUL is only possible when the same conditions as for classification problem formulation are valid - there is enough data of pre-failure observations. While normally, the SM fabs are operating under failure preventive measures, meaning that PvMs are regularly performed leading to

unnecessary equipment interventions. Therefore, corresponding data usually consists of "normal" schemes of equipment functioning until it is stopped before actually reaching crucial degradation in its performance.

Thus, when dealing with essentially faultless data sets, the most optimal solution is to consider learning ML model to estimate the future values of some quantities that characterize a system progression in case the tool is functioning in "normal" conditions. In fact, many processes exhibit inevitable steady drifts in nature because of the gradual wearing-out phenomena or build-up of material on the components of the tools [16]. That is, the process drift might be caused by the process itself or induced by the process tool. It results in variations in the fabrication outcomes from wafer to wafer. Hence, the process drift must be estimated. Since the SM usually suffers from a high level of nonlinearity, in recent years artificial neural networks have evoked great interest in the areas of process modeling because of their ability to learn complex nonlinear functions [9, 16]. In this direction, recent research demonstrates that CNNs are of the best for sequence modeling [8]. Then, several methods are available in the literature to analyze the predictive outcome with the goal to identify anomalies, like analysis of the residuals [52].

Another way, proposed in [18, 57], is to train AE with data that represents normal system dynamics, which learns how to compress and reconstruct this data. Then, the processing of anomalous data with the trained AE results in a reconstruction error analysis.

2.2.2 Virtual Metrology (VM)

The VM as a novel method, was introduced in 2005 [20] for Wafer-to-Wafer (W2W) control enhancement. It was defined as a correlation model between tool historical data (like temperature, power, flow rate, pressure, optical emission spectrum, and plasma impedance; initially collected for equipment failure detecting problem) and acquired properties of wafers (e.g. thickness, depth, critical dimension of a deposited layer) after the corresponding process is complete. Experimentally on simulation data of a shallow trench isolation deposition process the authors estimated deviation improvement up to 65% from using VM. The first VM correlation method was tested on data from solely one specific process and it triggered different teams to investigate the methodology on different industrial case scenarios.

Next, in 2007 VM mathematically was defined as a regression problem [37, 35]. In [37], authors were comparing Multiple Linear Regression (MLR), principal component regression (PCR), and partial least squares (PLS) methods in solving VM problem for lithography and plasma etch processes. While in [35], authors suggested using Radial Basis Function Network (RBFN) in VM scheme for the chemical vapor deposition process. Later in 2009 a study [30] was published again for the chemical vapor deposition process where authors showed that Feed Forward Neural Networks (FFNN) with sigmoid nonlinearities outperform MLR and RBFN approaches in metrology predictability. While following the work [45] in 2010 shows that gaussian process regression (GPR) can outperform a very good result of ANNs. The same result was later obtained for chemical vapor deposition process metrology predictions in 2014 [73], also compared to MLR and least absolute shrinkage and selection operator (LASSO) results. And the same year, another study [54] shows compatible results of support vector regression (SVR) for VM for chemical vapor deposition process monitoring.

The most recent research on VM includes also Deep Neural Network (DNN), like in the [34] of 2021 the authors argue that traditional ML methods, including FFNN, require feature selection process is required to extract the predictors and their performance improvement has its limits, while if CNN can replace FFNN in the VM system, not only the issue of time and labor consuming for feature selection can be improved, but the prediction accuracy can also be enhanced.

2.2.3 A Brief Review of Existing ML-based Approaches for PdM and VM

This Section is engaged in introducing the main concepts behind the most common approaches that are proposed in the literature to solve PdM (see Section 2.2.1) and VM (see Section 2.2.2) tasks.

Linear Regression

Simple and multiple linear regression (SLR and MLR respectively) are the most basic and consolidated modeling tools for explaining a response variable (regressor) from one (SLR) or more (MLR) numerical and/or categorical explanatory variables (predictors) by fitting a linear equation on the observed data

$$\hat{\mathbf{y}} = \mathbf{w}\mathbf{X}, \tag{2.8}$$

where a coefficient vector $\mathbf{w}$ is estimated using the least squares method:

$$\min \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{2.9}$$

SLR and MLR have excellent and well-studied theoretical properties, besides being very intuitive in explaining a resulting model. Such fast models define the simplest relation between input variables and target. Of course, they are limited in all problems where a nonlinear relationship between the regressor and the predictors is expected. Furthermore, MLR can be used only in a context where the number of predictors is much less than the size of the available samples. Otherwise, due to intrinsic illconditioning arguments, the variance of the estimated coefficients grows up to make them meaningless. Additionally, in the case when some explanatory variables show high correlations, there is a high chance that the performance of MLR can be impacted by a problem called multicollinearity. Then, the estimated $\mathbf{w}$ can be very unstable, which leads to poor predictions of the response variable. A way to face illconditioning and multicollinearity is to select a small smaller subset of predictors, for example, with stepwise regression or LASSO.

Support Vector Regression

Support Vector Regression (SVR), unlike most linear regression models, operates to minimize the L_2 norm of the coefficient vector under additional constrain for error margin ε that allows defining an "acceptable" level of error to fit the data. Moreover, the regularization

term for the values outside of ε is introduced, which is controlled by hyperparameter C. As a result, the objective function is the following:

$$\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}|\xi_i|$$

under constraints:

$$|y_i - w_i x_i| \leq \varepsilon + |\xi_i|$$

Additionally, the kernel function can be applied to transform the data to make it possible to fit with a linear model. The kernel functions exist of different types, but in this work, the Radial Basis Function (RBF) is used:

$$K(x_i, x_j) = \exp\Big(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\Big)$$

Ensemble Trees Models

The algorithms based on DTs non-parametric supervised models are of big interest in the SM modeling tasks as they are simple and can provide stable results and perform well even for small data sets.

Random Forest (Random Forest) combines a multitude of decision trees - "weak" learners, at learning time. Each tree is trained independently in parallel on the randomly subsampled small set of the data, so that it allows for improved variance and provides an unbiased estimate of the generalization error. Generally, Random Forest is considered as highly accurate estimator in many tasks; it can successfully handle high-dimension feature input and a large proportion of missing data. However, it has a risk of overfitting for especially noisy data sets, and in case categorical data with many different levels are included Random Forest has an “absent levels” problem (some categories may not have their representatives in the random subset). Therefore, for a more efficient use of categorical variables, the next ensemble method is considered.

In contrast, Gradient Boosted Decision Trees (GBDT) relies in making predictions on a combination of decisions trees that are trained sequentially one at a time, in a way that every new one is trained to reduce the error of the ensemble of previous learners. Specifically, at a step k the existing k models form the following predictive function:

$$f(x) = f_1(x) + f_2(x) + ... + f_k(x),$$

where f_i,$i \in 1, .., k$ is the decision tree estimator added at a step i. Then, when the next decision tree f_{k+1} is added to GBDT model, the k existing “weak” learners are fixed and left unchanged, while the new one is trained to reduce the error of the updated ensemble - it is trained on the following set:

$$S_{k+1} = \{(x^i, y^i - \sum_{j=1}^{k} f_j(x^i))\}.$$

Accordingly, boosting is an optimization approach that aims to minimize a loss of the model by adding weak learners using a gradient descent like procedure. Trees, in turn, are

constructed in a greedy manner, choosing the best split points based on purity scores like Gini or minimizing the loss.

One of the advantages of this method is that it can manage highly correlated features. Moreover, it also can manage missing values, which means that observations that have one or several parameters entirely missing are kept for training.

Categorical Boosted Regressor (CBR) is a new gradient boosting algorithm that successfully handles categorical features and takes advantage of dealing with them during training as opposed to preprocessing time. Another advantage of the algorithm is that it uses a new schema for calculating leaf values when selecting the tree structure, which helps to reduce overfitting.

Artificial Neural Networks

The ANN is a model exploiting layers of weighted neuron connections aggregated with the nonlinear activation function to generate the next layer of neurons. The first layer of neurons is formed of the input variables to be fed to the model. Then, if g^{l+1} denotes an activation function to create the next $l+1$ layer in the ANN, then one neuron x_j^{l+1} of this layer is computed as:

$$x_j^{l+1} = g^{l+1}\Big(\sum_{i=1}^{N^l}(w_{ji}^{l+1}x_i^l + b_j^{l+1})\Big)\,, \forall j \in \{1, ..., N^{l+1}\} \tag{2.10}$$

where N^l is a number of neurons x_i^l in the prior layer l, and w_{ji}^{l+1} together with b_j^{l+1} refer to the weights and bias respectively that are to be learned from the data. Eventually, the final layer of the ANN is defined by the target. The learning of the ANN is performed iteratively by updating the weights to minimize loss function using gradient descent algorithm.

In the past decades, ANNs gained much importance in fault diagnosis and virtual metrology tasks as they provide a good approximation of the nonlinear relationship between predictors and regressor. Besides, being applied on the feature space inputs, there exists an active investigation of ANNs like CNNs and LSTMs that use the temporal data that hold the most abundant information as an input data. Then, CNN automatically extracts subtle yet important features from temporal data through the convolutional and pooling layers. After flattening these features to get the prediction result. While LSTM is capable of learning long-term dependencies of the input sequences to estimate the target outcome.

2.2.4 Performance measures

Throughout the project, the Coefficient of Determination (R^2) has been a main error indicator to measure the performance of the predictive models and to compare the approaches with each other. Given a regression problem with y_i being the real value of a target variable in the set of size N, the Coefficient of Determination is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}, \tag{2.11}$$

with $\hat{y}_i$ being an estimate obtained by a regression model of choice and $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$ being an average of observed target values. Stated in other words, if the variable y to be estimated has a variance due to noise, the R^2 score tells how much of this variance is explained by the employed model with respect to a simple model where the estimate is just given by the average value $\bar{y}$. Then, $R^2 = 1$ only in case when the predictive model perfectly fits all data - $y_i = \hat{y}_i$ for all $i = 1, ..., N$. When the regression model gives as a solution the average value of y, ($\hat{y}$), then $R^2 = 0$, and no variance is explained by the model with respect to the average. It is possible to observe that R^2 could even assume negative values, despite its very definition, when the solution of the model is worse than the constant estimate given by the average value.

Accordingly, as closer the R^2 score to 1 as better the model is. However, it is to be mentioned that a value $R^2 = 1$ could be misleading when the problem setting is affected by a phenomenon called oversmoothing. In this case, an excess of fit, due in general to the use of many variables and of a data set both for estimating the regression model and R^2, makes that once computed on a newly, never seen by the model, set of data, it collapses to low values.

Therefore, the Coefficient of Determination maybe not used as a stand-alone indicator of the effectiveness of the model, but together with other performance measures, like Root Mean Square Error (RMSE) and the Mean Absolute Percentage Error (MAPE):

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}, \tag{2.12}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\%, \tag{2.13}$$

in order to provide an additional evaluation to quantify the performance of ML models.

2.3 Multi-View Learning with Missing Data

2.3.1 Learning with Multiple Data Modalities

Nowadays a big variety of modern technologies provide the possibility to collect observations that corresponds to the same phenomenon from multiple sources. More information we have - the more accurate predictions for this phenomenon we can make. This idea leads to the growth of interest in multi-view learning algorithms during these past few years.

Many advances have been made on both theoretic and algorithmic sides [11, 33]. The three main families of techniques for (semi-)supervised learning are (kernel) Canonical Correlation Analysis (CCA), Multiple Kernel Learning (MKL) and co-regularization. CCA finds pairs of highly correlated subspaces between the views that is used for mapping the data before training, or integrated into the learning objective [6, 28]. MKL considers one kernel per view and different approaches have been proposed for their learning. In one of the earliest works, [7] proposed an efficient algorithm based on sequential minimization techniques for learning a corresponding support vector machine defined over a convex nonsmooth

optimization problem. Co-regularization techniques tend to minimize the disagreement between the single-view classifiers over their outputs on unlabeled examples by adding a regularization term to the objective function [60]. Some approaches have also tackled the tedious question of combining the predictions of the view-specific classifiers [68].

All these techniques assume that the views of a sample are complete and available during training and testing. In fact, in real-world multi-view data collections missing observation occurs for single or multiple views, so that the corresponding modality is incomplete or missing entirely. Often researchers address this issue by including in the analysis only complete observations. In this way, case deletion methods (i.e., instances with missing elements are removed) are among the simplest approaches, but they may potentially lose some valuable information in the data, and as a result, we get biased and low-quality predictions. Instead of simply dropping instances lacking on information, a more convenient strategy would be to replace the missing values.

2.3.2 Incomplete Views

Various traditional imputation strategies have been successfully applied to complete partially missing, data on a view, which include statistical methods, like imputation with a mean, median, extra value substitution, as well as expectation maximization, full information maximum likelihood and multiple imputations approaches; and linear, polynomial, backward/forward, padding interpolation methods. All of them have been largely adopted in literature to impute missing values in different kinds of data. Another principled method of dealing with missing data is Multivariate Imputation by Chained Equations (MICE) [42], which adopts a chained equation over various iterations to estimate the missing values after an arbitrary initialization. However, a major limitation of these approaches is that they deal with a low percentage of missing values, thus the imputation accuracy decreases as the percentage of missing values increases. This drawback originates because these strategies do not always succeed in capturing non-linear relationships between observed and unobserved features.

Also, among the most common ML-based imputation models is KNN [49], which replaces missing values with an average of corresponding missing variables data among k closest in a space neighbor observations according to selected distance metric (usually Euclidean). While KNN works only with numerical data input and requires tuning of the parameter k, MissForest (MissF) [62] was proposed as a non-parametric missing values imputation method to deal with mixed-type of data, and was proven to be an effective solution in many types of applications. MissF, based on the Random Forest (RF) algorithm, is robust to noisy data and multicollinearity since tree-based approaches have built-in feature selection mechanisms. However, MissF does not consider any multivariate information among features to capture the missing mechanism.

Recently, many studies suggest completing missing parts from available data using Generative Adversarial Networks (GANs) [32]. As a matter of fact, GANs stand for the state-of-the-art solutions to distribution modeling tasks defined by a collection of data of any complexity. These models take their origin from the game theory and are formulated as a two players game formed by a generator and a discriminator neural networks. The generator, denoted as G, takes a random vector from a simple distribution, like Gaussian or

uniform, as an input and is supposed to produce a sample from a distribution defined by a given data, usually formed of complex objects like images, texts, or time-series. But it is not trained directly by qualitative comparison of distributions of real and generated samples. Besides, there is the discriminator, denoted as D, that determines whether a sample comes from the true distribution of the data or if it is synthetic objects that came from the generator. The classification error by the discriminator is the basis metric for the training of both networks. Accordingly, the discriminative network is trained to maximize the probability of assigning the correct label to both real samples and generated ones, while at the same time, the generative network is trained to minimize the probability of D labeling the synthetic sample by G into the "generated" class. Accordingly, a so-called two-player minimax game with the loss function $L(G, D)$ can be written as the following :

$$\min_G \max_D L(G, D) = \mathbb{E}_{\mathbf{x}\sim p_{data}}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_z}[\log(1 - D(G(\mathbf{z})))] \ . \tag{2.14}$$

In the direction of problems with partially missing data, several advanced GAN-based models were proposed to learn a distribution of interest even from lossy observations only [12, 41]. In this respect, in AmbientGAN model [12], a simulated random measurement function, or so-called corruption process, is introduced, so that the discriminator must distinguish a real incomplete observation from a simulated measurement of a complete generated sample - corrupted full sample by G. In this way, AmbientGAN assumes the measurement process is known or parameterized, which is not the case in general missing data problems. Therefore, authors of MisGAN [41] proposed to model the missing data process using one GAN for masks vectors generation indicating which entries of $\mathbf{x}$ are observed; and then they train the complete data generator adversarially by masking its outputs using generated masks and comparing to real incomplete data - a second GAN model.

Both AmbientGAN and MisGAN are proposed to learn a generation function to provide a sample from the distribution of complete observations. However, to solve the data imputation task, an additional setup, like learning one more GAN in the case of MisGAN model, is required, which make those model too expensive in terms of memory and time to train. On the other hand, an efficient model for missing data imputation called Generative Adversarial Imputation Net (GAIN) [76] proposes a much more compact model architecture. Its generator aims to impute missing components of the real incomplete sample and outputs a completed vector. Then its discriminator is modified to take a completed vector by G with the objective of determining which components were actually observed and which were imputed. The authors tested the method on various datasets and found that GAIN significantly outperforms state-of-the-art imputation methods.

2.3.3 Missing Views

As for the tasks where a view may be missing completely, recently, many studies have considered the generation of multiple views from a single input image using GANs and have demonstrated the intriguing capacity of these models to generate coherent unseen views. The former objective, first, was to propose the method for learning a joint distribution of multi-domain observations (usually images) with the data of unpaired samples of several

domains. Particularly, for a problem when two views are available, in Coupled Generative Adversarial Network (CoGAN) model [44] it was proposed to couple two GANs together by enforcing weight-sharing constrain for both generators and discriminators.

The next approaches were focused more on the problem of missing view imputation, which usually is considered as a domain transfer task when the missing view is aimed to be generated from the other one or more available views [43, 80, 75, 38, 36, 67]. In that respect, in [43], authors suggested an unsupervised image-to-image translation model UNIT based on CoGAN with incorporation to it the Variational Autoencoder (VAE) in order to map different domains into a common latent space that then both views can be recovered from. Then, unlike UNIT, authors of DualGAN [75] and CycleGAN [80] propose a more generic solution by dropping the assumption about the shared embedding space, by introducing one-to-one correspondence (bijective) mapping and dual learning. Particularly, they define one generator to learn a mapping from one domain to another, whereas a separate generator maps it back to the original domain, which allows to include an additional term to the value function standing for reconstruction error or cycle consistency loss in DualGAN and CycleGAN respectively.

Additionally, there were more works proposed in the same direction, like like to discover cross-domain relations with GAN, called DiscoGAN or a pix2pix [36] method for supervised setting when the model can be trained with paired data. Moreover, several approaches stated to outperform the CycleGAN, like Domain Transfer Network (DTN)[67] that differs by enforcing a consistency not only on the reconstructed sample but also on the embedding itself, or a missing view imputation with generative adversarial networks (VIGAN) model that additionally uses a Denoising Autoencoder (DAE) to learn latent spaces for each view for better missing view reconstruction.

The models mentioned above have shown their performance efficiency on the data sets consisting of two domains, while most of them may have limited scalability due to the necessity to learn the transfer function for each pair of domains. To address this limitation, in StarGAN [22], authors propose a framework to perform transfer methods to one domain from multiple different views using a single model, particularly using a single discriminator to control multiple domains. However, there are fundamental differences between image imputation and image translation, since transferring is still performed pairwise without considering the remaining domain data set. Therefore, a year after a Collaborative GAN (CollaGAN) [40] model was proposed, that shares the ideas of StarGAN, but solves the missing data imputation problem.

These are very exciting models, however, our learning objective for the part of missing views problem investigation in this thesis differs as we are mostly interested in the joint learning of the target label together with the multi-view data imputation task. The most similar work that uses GANs for multi-view classification is a multi-view bidirectional generative adversarial network (MV-BiGAN) [19]. This approach, first, introduces a conditional BiGAN model to learn the target label conditioned on one view or aggregation of multiple views, and second, it introduces a mapping function that allows mapping the set of non-missing views into a representation space.

Chapter 3

Predictive Maintenance

As anticipated in the Introduction, the outlay for maintenance and abrupt equipment outages in the SM industry (being a supportive expense that does not generate a profit) represent a quite big percentage of the total costs, since machinery components and skilled labor are expensive, while equipment downtime being an exponentially larger threat to production cost. The three major types of maintenance are reactive, preventive (PvM), and predictive (PdM), and the last one is the most attractive in terms of cost saving as well as productivity improvement. The objective of this chapter is to describe the methodologies and tools investigated as part of the PdM framework for the past three years of the MADEin4 Project, which is engaged in developing next-generation processes and metrology measurement tools, ML methods and applications in support of Industry 4.0, that stands for smart manufacturing for the semiconductor fabs.

3.1 Contributions

Within the MADEin4 Project we collaborated with the teams of European leading SM companies to analyze real fab data with the goal to propose a framework for predictive control and maintenance of process tools, particularly for the injection valve used for the Chemical Oxide Deposition (CDO).The study has commonalities with most of the maintenance problems, where heterogenous data coming from in-line equipment sensors and/or measurements on the manufactured device get too large and often too complex to be analyzed through human inspection. However, they have been specifically selected for the different nature of data (spatial vs. temporal data) and the different impacts on the manufacturing process (equipment downtime vs. cost-of-replacement).

During the project, first, we worked on understanding and visualization of ESD data collections from CDO equipment sensors, consisting of algorithms for automatic data-cleaning filtering, interpolation, outlier removal, and features extraction. We proposed a methodology for better process characterization, in terms of capturing necessary features to describe equipment performance degradation, that uses Gaussian Mixture Models (GMM) for multi-mode signal tracking.

Next, we investigated different statistical and ML approaches for data processing and modeling, including clusterization, classification, and regression models, with the goal to

propose a framework for CDO PdM problem. Their performance was tested and evaluated with the real tools sensory data, and the best practices - ones that reliably can reduce the mismatch between the machine tests and the maintenance system indicators in the given data - were selected and are described in this Chapter. Finally, a fab analytics and optimization tool based on ML algorithms was proposed and deployed in the fab for the evaluation with the new data (unseen during development), while many proposed approaches in the literature are missing a real-world validation [27].

3.2 Predictive Maintenance for the Chemical Oxide Deposition (CDO) Equipment

The (CDO) system is a key tool for the power production line. Currently, to keep under control the injection valve status, a direct parameters that can be measured and monitored does not exist. An indirect way to monitor the CDO system status was identified in a so-called interceptor variable, which can be used by APC as a conventional method for process maintenance with an automatic stop in case of value out of control (see Figure 3.1). However, the interceptor monitoring, unfortunately, fails in some cases, both with False Positive (FP) and False Negative (FN) estimations. The FP stands for the case when the interceptor value is out of control, but the CDO system is clean. Then, FP causes an unnecessary maintenance cost. On the other hand, the FN refers to the case when case the interceptor value is in the "normal" range, but the injection valve is clogged. Then, FN causes, even worst, production losses leading to the discharge of a big amount of processed wafers under a corrupted system state. In order to reduce the mismatch between the machine tests and current maintenance system indicators (interceptor) in the given data, within the MADEin4 Project it was proposed to develop a more robust and reliable ML based PdM approach relying on the exploitation of any information already present in the system (such as gas flow, chamber pressure, throttle valve steps, process temperature, foreline pressure).

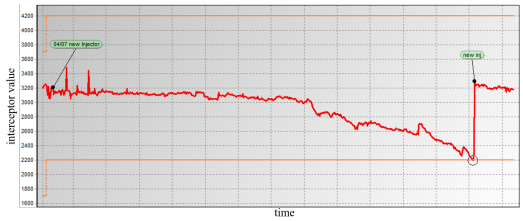


Figure 3.1: APC of interceptor monitoring (by STMicroelectronics).

In particular, we propose to build a framework for modeling process drift from its initial state that was observed after the latest Preventive Maintenance (PvM) event in order to detect deviations from normal conditions. To perform accurate prognostics, two important conditions must be satisfied: accurate characterization of the current system state and a model which describes the progression of the characteristics. Accordingly, our framework consists of two parts: features extraction and modeling, that both leverage several state-of-the-art ML methods integrated in a rather complex framework. Finally, predictions are supposed to undergo analysis and comparison with actual observations from the sensors with the goal to generate error warnings in case of detected anomalies, to consider scheduling maintenance on the tool. The ability to predict such maintenance events reflects the capability of building

dynamic maintenance schedules for yield optimization during manufacturing, which justifies the research effort.

3.3 Statistical Methods for ESD Feature Engineering

Any PdM strategy takes as an input raw Equipment Sensory Data (ESD), which is a collection of time series of different sensory parameters records as shown in Figure 3.2. Then, different methodologies have been explored, which aimed at detecting the best features of ESD indicating the upcoming of failure of the injection valve. As well in this study, we focused on extracting features from the raw ESD signals, in order to have a compact and uniform representative of as much as possible information on the process state.

Common techniques for process characterization in the SM data collections are described in Section 2.1. They propose considering methods for feature extraction (traditional statistical, signal decomposition, or representation learning) on a wafer basis of granularity. Accordingly, one observation of the ESD is one wafer cycle of sensory parameters records that are a subject for feature extraction methodology. However, in the PdM use case (when system state estimation is not necessarily to be done after each processed wafer) we argue that this approach may bring systematic noize or variation to the process characterization.

3.3.1 ESD Data Windowing Strategy

In fact, there exists a natural variation of the ESD signals depending both on chamber and recipe, despite the equivalence of measured sensory parameters among chambers as well as recipes. Particularly, one wafer's ESD records are aligned and have equal

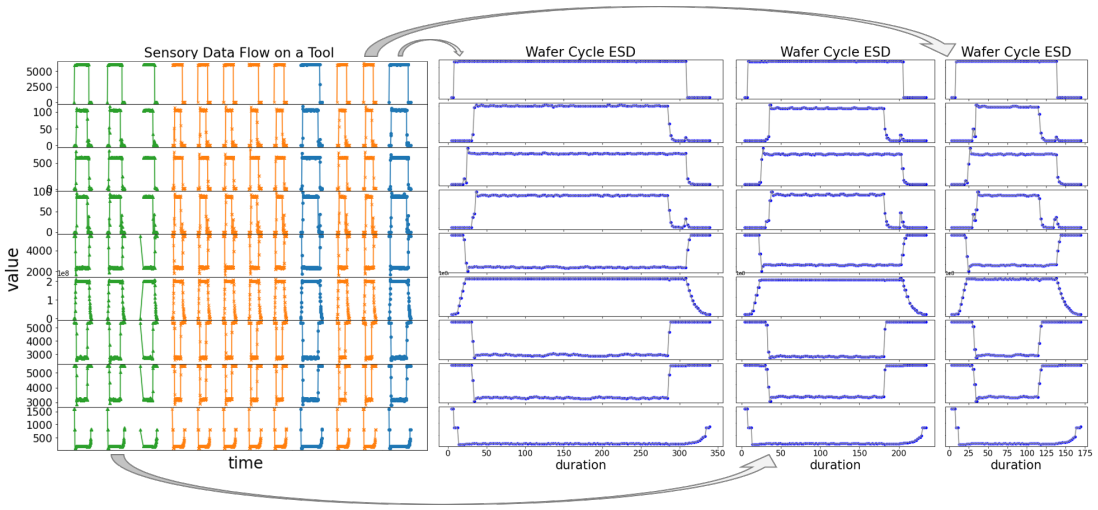


Figure 3.2: The very left plot is a fragment of several ESD wafer cycles that consist of recordings of 9 different sensory parameters collected from 12 subsequently processed wafers on the CDO tool. Different colors on the very left plot represent different recipes that the corresponding wafers were processed with. The rest of the plots shows a duration variability of the wafer cycles depending on the choice of their recipe.

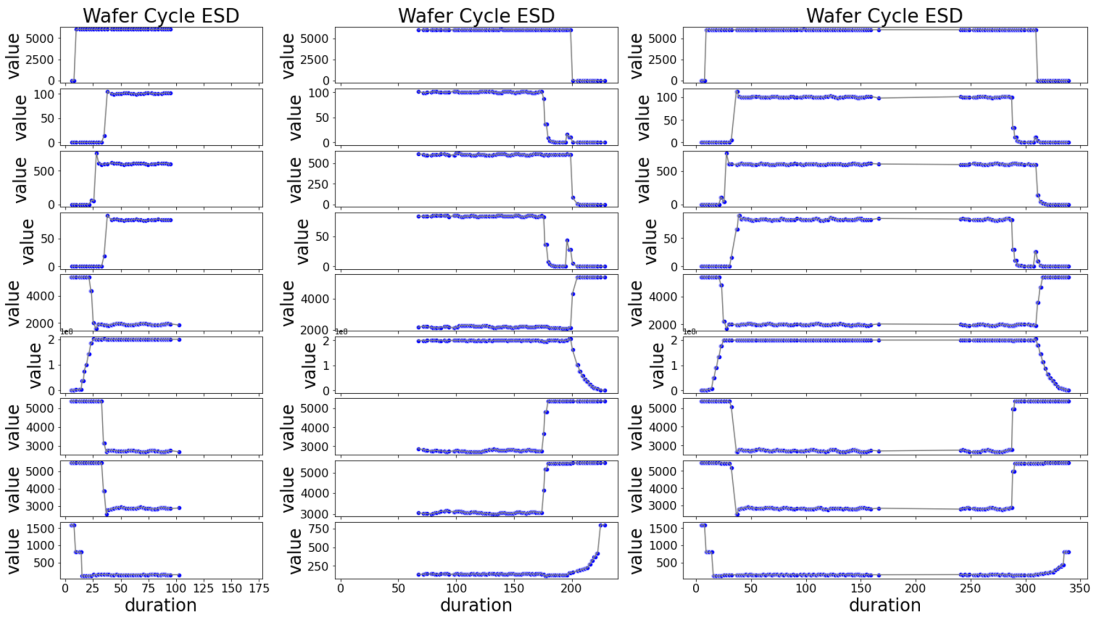


Figure 3.4: Variability of incompleteness (missing data) of wafer ESD cycles.

duration, while the duration of sequences of all the wafers generally depends on their recipes. To show such a variability, Figure 3.2 gives examples of several parameter records that were recorded while processing three different wafers on the same equipment but with different recipe settings. Additionally, there exists a variation in lengths of wafer cycles ESD recordings of the same recipe (Figure 3.3), since the recording tool may have a fluctuating sampling frequency due to natural tool variability, as well as incomplete ESD wafer cycles may occur accidentally (Figure 3.4). Then, all of the factors mentioned above cause a huge variance of the extracted features.

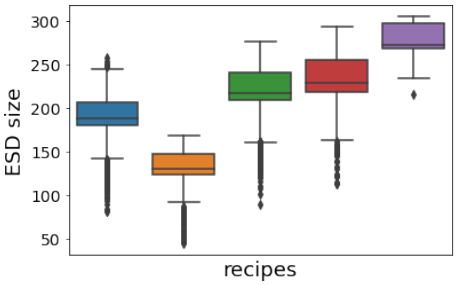


Figure 3.3: Variability of the wafer cycle lengths (total number of sampling points) of the ESD data.

Therefore, we operated the ESD data with a windowing strategy to remove all the possible sources of systematic variations retaining only random variations due to the natural variability of processes in order to generate better quality input for the modeling part. Accordingly, the granularity of the ESD is modified, so that one observation becomes a window of fixed size (based on time, or a number of records, or a number of wafer cycles) of considered time series, rather than one wafer cycle ESD. The size of windows is a tunned parameter which is a compromise between two objectives: 1) provide predictions with as small as possible frequency, and 2) have enough data in the window to generate an accurate characteristic for the tool state with less variance as possible.

3.3.2 Gaussian Mixture Models (GMM) for Multi-mode Signal Tracking

In order to build a set of features that characterize tools functioning conditions, several feature extraction methodologies are applied to the ESD data of considered windows. In this work, statistical features (mean, std, median, min, max, skewness, kurtosis, and other characteristics) are computed on a distribution of subsets of parameters' values, instead of full parameters time series of the windows. Specifically, we assume that the distribution of parameter's values observed in windows is a mixture of a finite number of several distributions (Figure 3.5c), that reflect the different operating modes of the CDO system (Figure 3.5b). Then, similarly to [13, 63], we propose to treat modes, or so-called phases, separately which will allow for changes in each of the CDO operating stages to be tracked over time.

To do so, we used a Gaussian Mixture Models (GMM) - a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters – for so-called clustering of the parameter's values observed in corresponding windows. Specifically, at each iteration for every parameter in the ESD collection, the best-fit mixture of Gaussian distributions is determined which best describes the underlying distribution of the parameter's values within the window. Then, the features are extracted for each of the identified clusters defined by identified Gaussian distributions from the found mixture. And as a result, this approach gives an automatic way to split a signal into multiple modes and extract the characteristics of each of the modes separately.

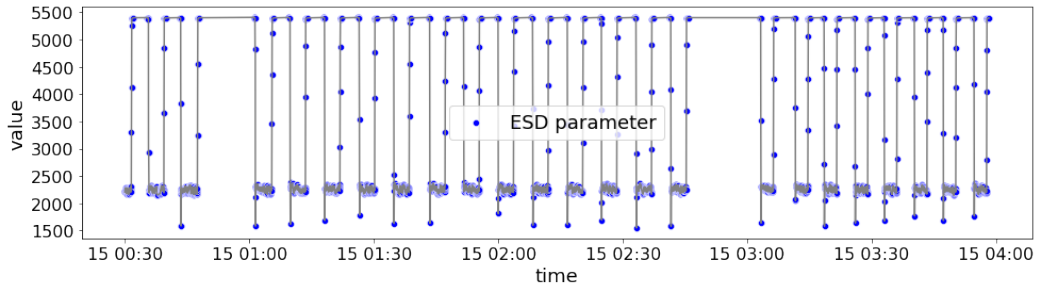
Figure 3.6a shows an example of a progression of one the parameters from the ESD through full time span of the CDO system functioning starting from one PvM until the next one. While, Figures 3.6b and 3.6c show how well this progression is represented through statistical features extracted traditionally and by mode splitting respectively. Accordingly, the traditional feature extraction way shows much higher variation than the mode-wise features extracted after the GMM clusterization.

Finally, in order to reduce a distribution difference/shift associated with different kinds of events on the equipment – maintenance, and tool recalibrations, we propose to center the extracted features accordingly. In this work, we operated the progression of extracted features shift through their delta values regarding the initial state of the system after its last maintenance event (PvM), as depicted in Figure 3.6d.

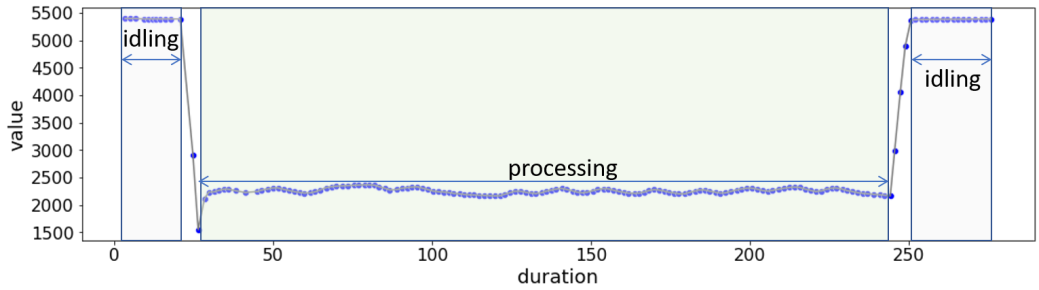
The advantages of the selected approach (that includes data windowing, GMM modes split and centering with respect to PvM events) for raw data processing are the following: the methodology is not affected by the inconsistent length of the sensory records, and it does not require parameters time series to be equispaced; extracted characteristics were proven to accurately highlight trends in data with the minimum noise observed.

3.4 Modeling Approach

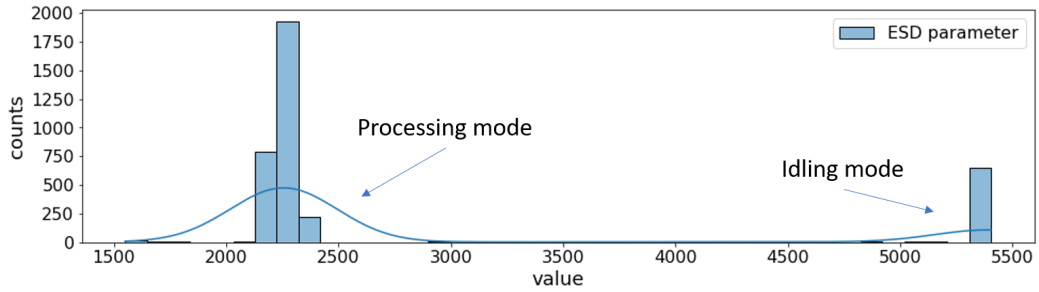
As anticipated before, in this work the predictive model is built to estimate the process parameters features drift from their initial values that were extracted after the latest PvM



(a) A 4 hours window of ESD recordings of one of the sensory parameters.



(b) Zoomed wafer cycle ESD of the parameter sequence depicted above.

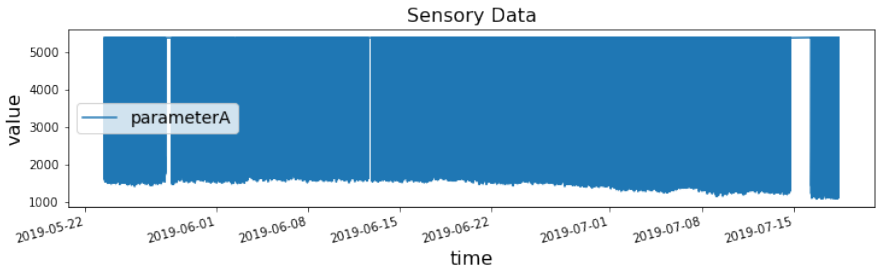


(c) Distribution of values of the considered parameter inside the 4 hours window (see Figure 3.5a).

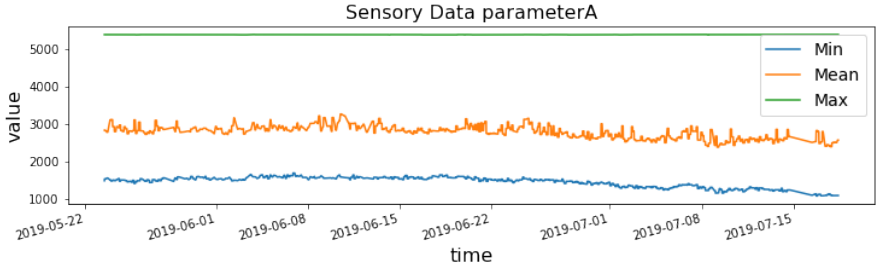
Figure 3.5: Analysis of values of one of the ESD signals in the time-window of 4 hours.

event. Therefore the historical training data timeline was divided by the PvM schedule. Then, there were identified the parameters that do not change (in terms of modes' distribution drift) in-between two sequential PvM events; and the rest of the parameters were taken for further processing (like one that is shown in Figure 3.6). Selected parameters then are those that have any king of progression/evolution of their process characterization features in-between two sequential PvM events which we assume can be prognosed (in case of normal tool functioning conditions).

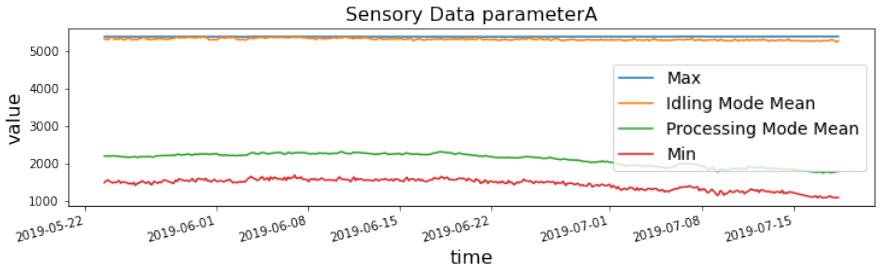
The core approach for a predictive diagnostic of the CDO tool that we used in this work was selected as Gradient Boosted Decision Trees (GBDT), which is an ensemble-based method for regression and classification tasks, where trees are trained in sequence over residuals of the loss function. Trees, in turn, are constructed in a greedy manner, choosing the best-split points based on purity scores like Gini or to minimize the loss, and the main



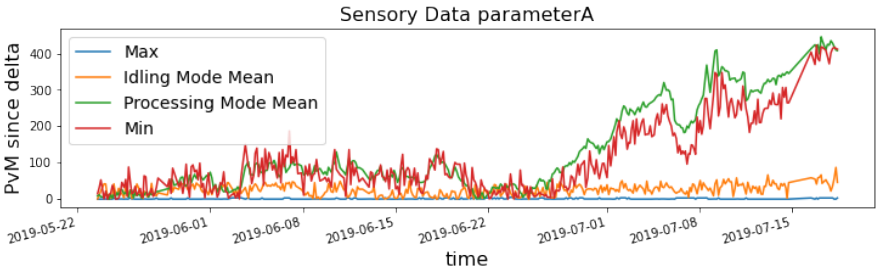
(a) Display of the ESD raw values of the *parameterA* signal as a function of time in between the span of one machinery processing cycle.



(b) Display of the *parameterA* signal (from Figure 3.6a) decomposition into the traditional statistical charachteristics.



(c) Display of the *parameterA* signal (from Figure 3.6a) decomposition into the modes' (identified by the GMM) charachteristics.



(d) Progression of a deviation of the *parameterA* signal modes from their initial setting.

Figure 3.6: The progression of the ESD *parameterA* and its features decomposition withing two consecutive PvM events.

cost of GBDT is the construction of decision trees and the most time-consuming part is finding the best-split points for each node. More details about the approach are provided in Section 2.2.3.

The advantages of the selected approach of modeling are the following: it was experimentally proven to be an accurate estimator for considered use case (shown next in Section 3.5); it deals effectively with different types of data (numerical and categorical) and can successfully handle high-dimension features input; it can provide a good estimate even in case of missing data, which is a common issue of the manufacturing collections of records; it is interpretable, and more importantly, it provides an explanation of the contribution to the prediction of every variable in the features set.

3.5 Experimental Results

The proposed framework has experimented with the real SM data collection provided by STMicroelectronics, which consists of the ESD records of several chambers of the 9 months time interval, which includes five full machinery processing periods without interruptions, that we denote in this work as a system running cycle. One out of the five was ended because of equipment failure which the interceptor monitoring, unfortunately, failed to detect/prevent.

Importantly, for the training phase of the developed framework requires a collection of records that covers several equipment running cycles - from one preventive maintenance till the next one, that has a successful quality check. It is principal to create a diverse collection of “normal” tool functioning conditions for better modeling.

Thus, for the experiments, all four "healthy" machinery running cycles given in the data for this project were selected for model learning. In order to have a correct setup that avoids overfitting, the following cross-validation scheme was adopted: the former data set of "healthy" machinery running cycles was split by 4-fold group cross-validation as training and validation, meaning that all records of one cycle out of four formed the validation subset and the remaining ones were assigned to the training subset. The 4-fold cross-validation is used in the framework to train the model on the training folds and then to evaluate it on the validation ones for choosing hyperparameters. In the end, once the hyperparameters have been tuned, the regression model is again estimated once on the full training+vaidation set using these hyperparameters. Finally, this last model predicts the progression of the selected process parameters features for every next 4 hours window of tool functioning on the test set, which is independent of observations used to train the model and tune its hyperparameters, and is one that has failure occurrence in it.

In a predictive phase, the estimated values of the drift are predicted and compared with the numbers that are actually observed on the tool. The comparison is performed by measuring an absolute error for every prediction-observation. In order to estimate if the error is within the acceptable range, we performed a statistical analysis of the errors across the entire time span of the training data, and decided to normalize the observed error by dividing them by the average absolute error on the training. Then, in case the normalized error is bigger than a threshold equal to 1 - a warning is raised; in case the error is bigger than threshold 2 - a failure alarm is raised; otherwise - the tool is functioning normally.

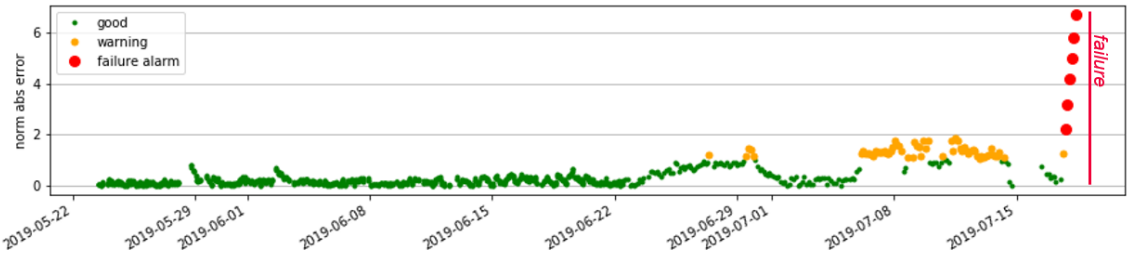


Figure 3.7: Display of the absolute difference between the prediction values by the developed PdM framework and the actual observations on the testing equipment. The reported testing time span is one cycle of the machinery functioning period, independent from training observations, that has a failure record at its end.

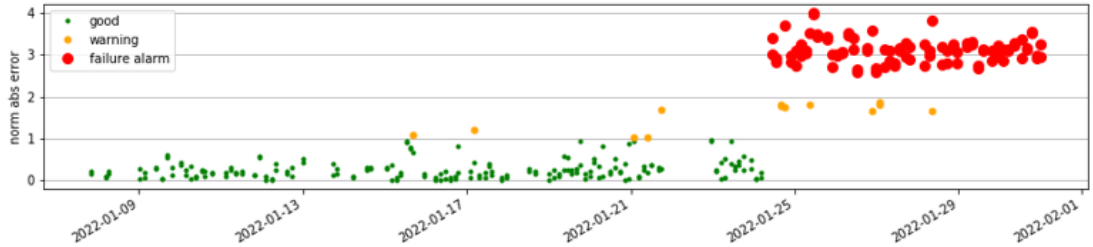


Figure 3.8: Display of the absolute difference between the prediction values by the developed PdM framework and the actual observations on the testing equipment. The reported time span is one cycle of the machinery functioning period in the real fab test environment, independent from training observations, that has a failure record at its end.

Finally, Figure 3.7 displays the performance of the developed PdM application on the test set. The result shows the ability of the framework to alarm the upcoming equipment failure a day ahead which may give time for intervention to the process preventing the manufacturing outage.

3.6 Discussion

At the beginning of the MADEin4 project, maintenance of the CDO equipment was cyclically performed according to the rigid scheduling and conventionally developed process alarm system. However, the offtimes PvMs and not predicted machinery faults result in a limited availability efficiency (AE):

$$AE = \frac{Uptime}{Operation\ Time}.$$

Within the project, we proposed the practical methodology for the CDO process characterization, which combines the GMM clusterization of ESD records and traditional features extraction technique based on distribution summary statistics, allowing to define the informative predictors for the modeling of the natural system state progression to identify its deviation from the normal tool functioning.

The developed framework was adapted to STMicroelectronics fab infrastructure and tested in a demo environment. The results, presented in Figure 3.8 demonstrate the efficiency of the proposed approaches to catch in advance the upcoming issue and consequently to increase the AE by 30% with respect to the pre-MADEin4 value. Increasing the AE and anticipating unscheduled downtime it is possible to preserve potential electrical drift on devices and improve the products' yield. Furthermore, the developed IT architectures can be reused for the next developments in this field, which can be:

- An investigation of a transfer learning approaches to enhance the generalization through making more accurate predictions when the new processing cycle is started;
- Better use of maintenance (both PvM and PdM) data that affect time series of ESD for better identification and characterisation of the process drift.

Chapter 4

Improved Semiconductor Process Characterization Using Virtual Cross Metrology

In this Chapter, we introduce a multi-view methodology to combine process, and design information within a single Virtual Metrology (VM) framework for metrology modeling that utilizes ML techniques to learn about the process and design contributions affecting manufacturing. This chapter is based on the following papers [Ana21, Ana22a, Ana22b].

4.1 Motivation

As anticipated in the introduction, the Virtual Metrology (VM) is a key enabler for productivity enhancement as its goal is to exploit a piece of information already present in the system (eg. equipment sensory data, tool settings, recipe specifications, design information) in order to infer the value of a costly or unmeasurable variable that is important for characterizing the production quality, without physically conducting the measurements. In state-of-the-art literature, many linear and non-linear ML predictive methods are investigated on the efficiency to solve VM tasks in the SM processes regarding different use cases, such as chemical vapor deposition, factory-wide control, etch depth prediction, and more (described in Chapter 2.2.2). However, the introduction of new, usually more complex, individual operations requires considering every new particular use case at hand, as it may require different or adapted to the use case VM framework approaches.

For the past three years, the MADEin4 Project has developed next-generation processes and metrology measurement tools in support of Industry 4.0 aka Smart Manufacturing for the semiconductor fabs. As modern processes have become more compound, more individual operations are needed to manufacture each level in a semiconductor product. Thus, in order to improve the precision of VM strategies, the ability to leverage the wealth of information to fully characterize the process, and determine the value of new measurements, has become an ongoing activity across SM companies participating in the MADEin4 Project. Accordingly, it allowed the delivery of practical ML applications for the VM framework consisting of the architectures for processes like data gathering, data pre-processing, and

data analysis in the semiconductor industry that are discussed in this Chapter.

4.2 Contributions

First, within the MADEin4 Project, we collaborated with the teams of European leading SM companies to analyze real fab data provided by engineers from STMicroelectronics, which is of a particular process of the advanced manufacturing, called Copper Electroplating Deposition (CuECD). In 1980-90s IBM's researchers proposed to utilize Cu - Copper, as a replacement for aluminum, in manufacturing electronic devices to improve their speed and performance [5], and since then it is widely used in the VM fabs. Our objective was to eliminate tests on wafers through measurements of thickness on CuECD by post-metrology control with VM predictions of the Cu deposited thickness, using data on process and equipment parameters measured during the ongoing process. Accordingly, we proposed a methodology for the CuECD process characterization. Moreover, we provided a technique for a design features extraction that helped to explain better the observed variance in the target.

Then, we investigated and compared different ML and statistical families of functions on their ability to fit system data for the metrology outcome. Particularly, we were focused on methodologies, like ensemble decision models, that are explainable and permit a determination of the main predictors to contribute to the estimated outcome, which can serve to assist the APC systems during manufacture. After, the best practices were selected to propose a fab analytics and optimization tool based on the ML algorithms for an analysis of highly diversified equipment sensory data together with design features that facilitates process modeling and management tasks in the SM fab. The developed framework was actually deployed and tested in the STMicroelectronics fab (in a demo environment), while many proposed approaches in the literature are missing a real-world validation [27].

Next, we worked on expanding the scope of traditional process modeling in SM by cross-process analysis. As product manufacturing is a sequential procedure of applying ordered processes to deposit new layers of features, one can use the precedent history to learn its impact on the current modeling target of interest. This was being done by analyzing the real fab data of the full wafer production cycle, provided by engineers from IMEC, and harnessing the ability to leverage the wealth of prior-to-the-moment information on wafers to be able to determine and extract value/ranking of current and past measurement that drive the outcome (metrology estimation). Accordingly, we introduced a methodology, called Virtual Cross Metrology VCM, that benefits not only from different types of measurements, but from dependencies between different process steps, in order to make processes more predictable and productive.

4.3 Virtual Metrology for Copper Electroplating Deposition (CuECD)

The work presented in this Section of the current Chapter, concerning the development of a framework for in-line controlling direct Copper deposition by electrochemical methods

for the manufacturing of semiconductor devices, was carried out on the data collected in the production lines of STMicroelectronics enterprise. These data are mainly guided by the experience of process engineers in assessing the condition of the wafers during the manufacturing process.

4.3.1 Problem Formulation

A CuECD system generally has several process units, called chambers. Individually, the chamber is where CuECD is performed for one wafer at a time. It contains a liquid coming from its related tank, and the deposition is performed once the wafer is inside in contact with the liquid while it rotates. Accordingly, each chamber is functioning independently and has its local schedule/frequency of maintenance, recalibrations, and metrology sampling, and therefore it can be considered as a stand-alone machine itself, and CuECD system may operate with some chambers being off process.

One machinery usually performs the deposition for many of the different products being produced in the fab. In this respect, the chamber is operating with different configurations to be set, called recipes, depending on what is the requested design of the wafer is currently in production. Accordingly, the expected outcome of the deposited layer is ruled by the design and the recipe. Overall, there are many different products being produced in parallel in the SM facility, and therefore wafers in a queue to the chamber usually require different recipes than the one being used prior to.

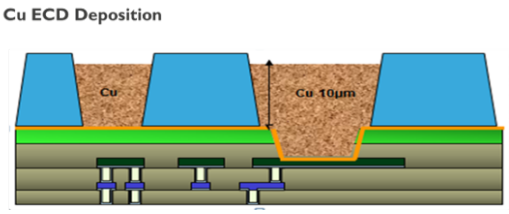


Figure 4.1: Scheme of a section of device after CuECD process.

In the fab production routine, wafers are traveling from one process to the next one in batches, called lots, normally formed of 25 wafers maximum of the common product type (design). While during CuECD, wafers of one lot are processed in parallel in all available functioning chambers. Once a wafer is processed with one of the chambers of CuECD equipment, it undergoes a quality inspection performed by measuring a thickness (metrology) on the just deposited feature layer (Figure 4.1), which is of great interest, as it helps to validate outgoing product proper functioning, as well as exploring and validating productivity enhancement opportunities. Due to the nanometer precision of the measurements and novel advanced technologies involved in the inspection, the thickness sampling for each wafer in a lot leads to a high cost of production and significantly increases the fabrication cycle time. Therefore, a few wafers (5 on average) are usually sampled for the inspection, and then are considered to represent the whole lot. However, such interpolation for the unmeasured wafers in the lot may cause undetected product defects, since the processing conditions of wafers within one batch are different (due to different chambers involved and their natural tool variability and disturbances).

The VM, in its turn, aims to provide more coverage at a lower cost by utilizing ML methodologies to learn from the available data on the system, like Equipment Sensory Data (ESD, which leads to an increased number of sample points that can be analyzed in order to have improved product monitoring or better process control. From a mathematical point of

view, the VM problem can be stated as a regression one, where metrology output (thickness) is linked in a function f with the process variables and other information available for the process and/or the product in the system, so that the estimated target metrology of i-th wafer w_i on the tool, $\hat{m}(w_i)$, is given by

$$\hat{m}(w_i) = f(\mathbf{p}^1(w_i), \mathbf{p}^2(w_i), ..., \mathbf{p}^P(w_i), \mathbf{u}(w_i)) \tag{4.1}$$

where $\mathbf{p}^s(w_i)$ refers to a vector of the parameter records of one of the sensors $s \in \{1, ..., P\}$ taken while wafer w_i was under the process treatment in the chamber, and $\mathbf{u}(w_i)$ that refers to any other available auxiliary information, like recipe or design - set of product specifications.

4.3.2 CuECD Data Processing

The SM data regarding the CuECD process that is present in the system for APC tasks is a collection of records and characteristics of different types and from multiple sources. Essentially there are two main modalities of the data given for the VM task. First is metrology data - thickness, which has to be predicted. And second, are the process time sequences that are measured during the growth of process on the semiconductor -ESD, and are used to build a predictive model from for the metrology outcome. In the present Chapter, such data is addressed using statistical terminology as predictors. Among such predictors also so-called ancillary variables are included, such as a recipe, design, chamber, and maintenance information. They are not measured during the growth of the CuECD process, but depend on the semiconductor to be produced. Table 4.1 provides a short description of different kinds of predictors available in the task, while further they are explained in detail, as well as their importance and preprocessing/treatment strategy in the developed framework.

4.3.2.1 Metrology

Since metrology measurements represent chip characteristics, it is expected that the whole die grid on the wafer would be sampled. For the same reasons that just a percentage of the wafers in a batch are measured, metrology is sparingly sampling the die granularity. Particularly in this work, on every sampled for the metrology investigation wafer there are only 5 spots, each located at one of 5 specific devices (chips), where the thickness is measured at. The selection of dies for the metrology measurements is driven by a goal to have full coverage for the variability of the outcome across the wafer (depicted in Figure 4.2), which exists and depends on the die/wafer treatment protocol of the particular process. It is common that there exists a dependency of the die metrology response to its distance from the center of the wafer. Hence, the devices for the metrology inspection normally are selected one from the center and a few at a certain radius, the same is observed in this work. We assume that measurements at the 5 spots are taken always at the same coordinates across different lots and wafers.

Then, if there are predictors to explain the metrology outcome variability at a die granularity available in the data collection, the VM task is being solved to predict the metrology for every die on a wafer. Exactly this kind of problem is being discussed in the

Data Group	Type	Description
`Process`	time series	• ESD formed of multivariate sequences (as a function of time) of values collected from equipment sensors for critical process parameters, like temperature, pressure, gas concentration, and more
`Recipe`	categorical	• recipe ID • any categorical feature to describe a recipe
	continuous	• duration • numbers to be set on the chamber to run the required process specifications
`Design`	categorical	• product ID
	continuous	• general properties from the layout design, which we know are characteristics that drive a specific process
`Chamber`	categorical	• chamber ID
`PvM`	categorical	• type of the latest PvM event (on chamber or total; scheduled or due to alarm; and more) • chamber/system state ("failure"/"success") prior to the latest PvM event
	continuous	• time elapsed since the latest PvM event • number of produced wafers since the latest PvM event • initial system/chamber state (by a set of sensory values) after the latest PvM event • maintenance indicators test results

Table 4.1: Description for the variability of groups of the Semiconductor manufacturing data and its types.

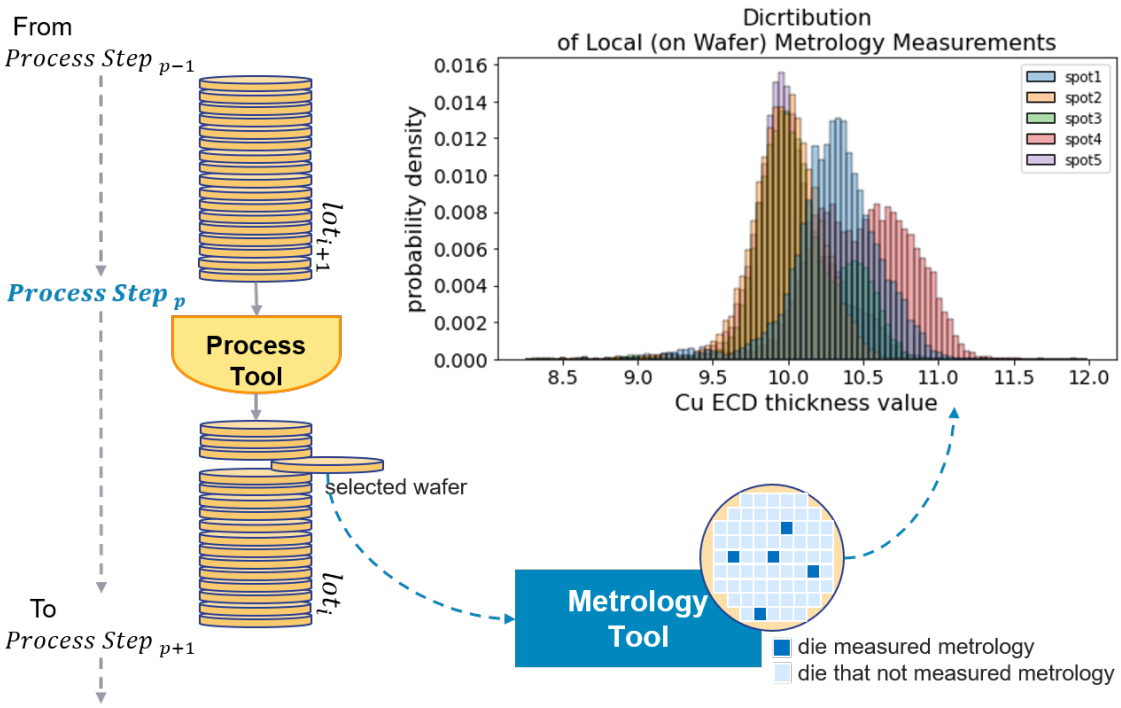


Figure 4.2: A diagram to describe a routine in the SM fab of wafers traveling through the production chain of process and how the validation of the processed devices is organized using metrology tools.

next Section 4.4 of this Chapter. While in the CuECD use case, the predictors are missing the local characteristics of the wafers, and therefore the target is defined as an average thickness (mean value over 5 measurements at the preselected spots).

In Figure 4.3 a distribution of the average thickness from the data collection used in this work is presented. It is possible to observe that the shape of the thickness distribution is not compatible with a Gaussian, which suggests the presence of more intricate components. It will be possible to see further that the thickness depends on recipe and product type, and its distribution highly concentrated on a few recipes and products produce the non-Gaussian distribution shown in Figure 4.3.

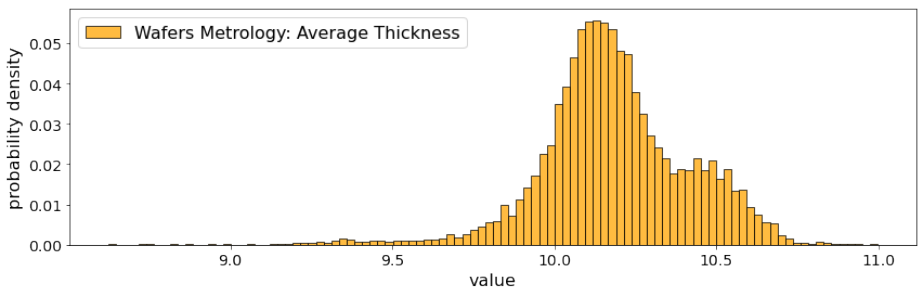


Figure 4.3: Histogram and distribution of average thickness (metrology data set).

4.3.2.2 Process Characterization

The core of the data of predictors consists of process sensory parameters records - ESD. Specifically, each chamber of the CuECD process machine is equipped with the same set of sensors that record numerical values of critical process parameters as a function of time when the chamber is running. If P is the total number of sensors, then corresponding process conditions at a time t (at a specific chamber) can be described as a P-dimensional vector:

$$(p^1(t), p^2(t), ..., p^P(t))^\top,$$

where p_s is the parameter value recorded from sensor $s \in \{1, ..., P\}$ at a time t.

Lets assume that wafer w_i was processed at the chamber starting at time $t_0(w_i)$ with a specific duration $\Delta t(w_i)$ (defines by a recipe), then

$$\mathbf{p}^s(w_i) = \{p^s(t_T(w_i))\}_{T=0}^{N(s,w_i)},\ t_0(w_i) \leq t_T(w_i) \leq t_0(w_i) + \Delta t(w_i),$$

where $s \in \{1, ..., P\}$ and $\mathbf{p}^s(w_i)$ stands for a collection of values of the parameter p^s that were collected from the sensor s while treatment of the wafer w_i. Then process conditions $\mathbf{x}_i$ of a single $i\text{-}th$ wafer in production is a multivariate time sequence denoted in the following way:

$$\mathbf{x}_i = \{\mathbf{p}^1(w_i), \mathbf{p}^2(w_i), ..., \mathbf{p}^P(w_i)\}.$$

Features Extraction

To assure the quality and effectiveness of the VM application it is necessary to perform a transformation of the preliminary process data $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ into a suitable form of input variables for predictive models, both for accuracy of the solutions and to avoid oversmoothing. Due to the following issues, as was previously anticipated, the timing of different parameters as a result of natural tool variability:

- time series are of different lengths
  $|\mathbf{p}^s(w_i)| \neq const,\ \forall i \in \{1, ..., N\}\ \forall s \in \{1, ..., P\}$;
- time series sampling is not equispaced and not aligned
  $t_{T+1}(w_i) - t_T(w_i) \neq const\ \forall T \in \{1, ..., N(s, w_i)\}\ \forall i \in \{1, ..., N\}\ \forall s \in \{1, ..., P\}$;
- dimensionality of features (average number of values in $\mathbf{x}_i$) is higher than number of observations
  $\frac{1}{N}\sum_{i=1}^N \sum_{s=1}^P |\mathbf{p}^s(w_i)| \gg N$;
- missing values may occur.

While missing values is an issue that is deeply discussed in the next Chapter 5, for now, we exclude the incomplete observations (that are missing completely the records of one or more sensory process parameters) from consideration. Then, one of the most popular methods for reducing the number of variates is to deal with features extracted from the time series instead of the original ones. Section 2.1 in the previous Chapter describes all the possible features (statistical and descriptive) that are usually used in different VM frameworks, which we also employed in this project. Such features are intended to catch

the main characteristics of the signal, also attempting to get basic local information on the sequence. While they could be used with all of the parameters $\mathbf{p}^s \ \forall s \in \{1, ..., p\}$, some "special" features were also investigated specifically to some parameters shapes presented below.

Overall in the CuECD VM task, the ESD collection of 10 different sensory parameters that are depicted in Figure 4.4 was given.

First, the shape of the parameters 9-10 (see Figure 4.4) implies an additional computation of their slope (angle).

Moreover, as from Figure 4.4, parameters 6 and 9 are potentially one of the most interesting as far as predictivity is concerned, because it shows the greatest variability among sequences. Looking at the plot in more detail, the process sequence is composed by one or more almost straight lines not consecutive but interrupted. Therefore some specific features have been extracted on the number of these straight lines, their start, and the jump between interruptions.

Finally, it is possible to see that the curves of parameters 1-6 are essentially a square inside the signal, with start, end, and duration slightly depending on the actual wafer/lot (besides recipe and product explained further). Specific features were extracted on so-called "jumps", on the values across "jumps" and their differences. To this purpose, a tool for accurate "jump" detection was also used as described in [66]. Moreover, the GMM features

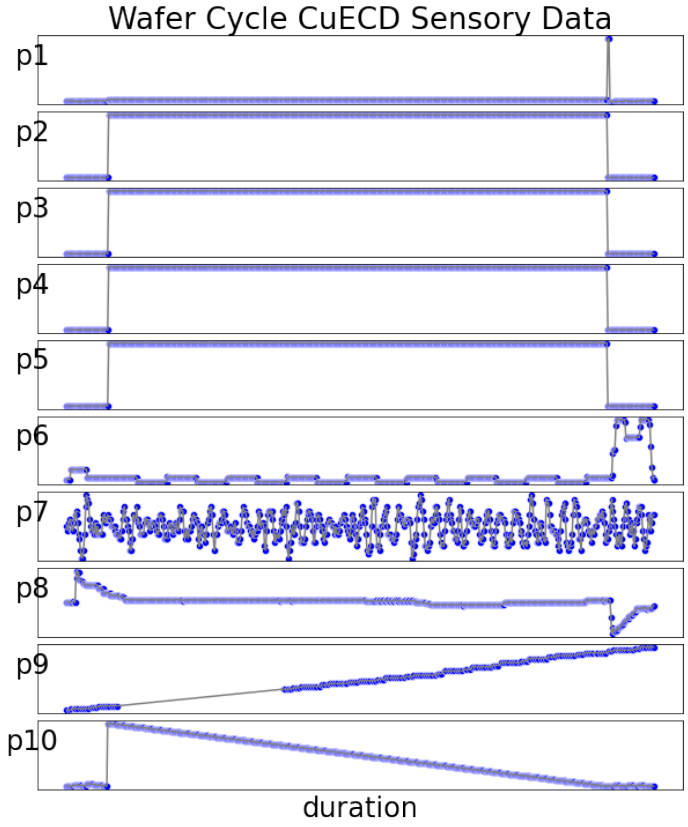


Figure 4.4: Example of a one wafer cycle ESD of the CuECD process that consists of recordings of 10 different sensory parameters.

extraction method proposed in Section 3.3 of the previous Chapter was applied, since the distribution of their values was assumed to be compatible with a mixture of Gaussians.

Features Selection

Some of the extracted features can measure similar quantities, therefore, can show high correlations. Then, there is a high chance that the performance of a predictive model can be impacted by a problem called multicollinearity. How to proceed with correlated features is a matter of choice and depends on many factors, starting from the regression framework. LASSO-type methods or boosted trees algorithms, for example, are immune to multicollinearity by nature, while linear regression can possibly be numerically unstable due to highly correlated predictors.

In all cases removing highly correlated variables reduces the feature set size, therefore saving computational time for running predictive models and possibly giving more accurate solutions. In this work, a threshold of the maximum admissible correlation to 0.9 (in absolute value) was set in order to identify strong positive or negative relationships between features. Accordingly, the highly correlated groups of features were identified and only one predictor out of the such group was selected for the final set of features used by the VM solver. The selection was driven by the best performance of the SLR model for the metrology outcome.

4.3.2.3 Design Characterization

How the presence of different product designs in the data influencing the distribution statistics of the metrology values is depicted in the Figure 4.5 created using the data collection available in this work consisting of 146 different designs. In practice, the ESD predictors may not be able to describe the variation of the metrology outcome across all of the designs, which is shown in Section 4.3.4. Therefore, "product ID" is usually additionally included as a categorical feature to the set of predictors to boos the precision of the developed VM framework.

Then, the presence of different product designs in the manufacturing fab is driven by the market request, and for this reason, we can expect significant variations in the ratio of different products in the collected data over different years. Moreover, even new designs normally are introduced in the production line that are not present in past years. In this respect, using "product ID" as a categorical feature may affect the generalization, because

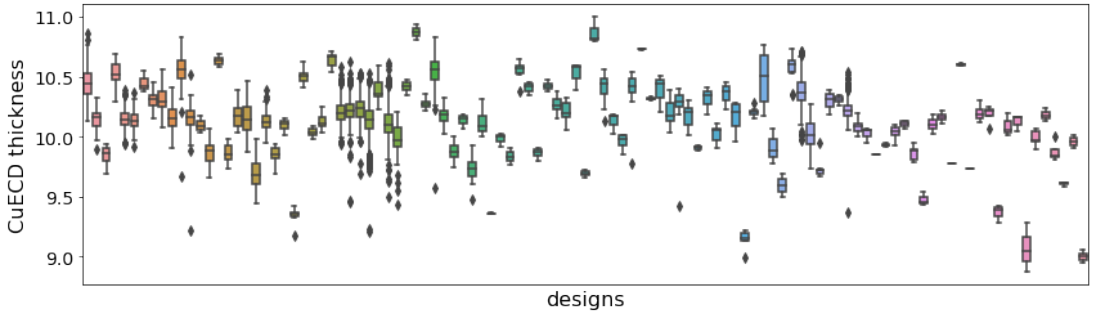


Figure 4.5: Display of the distribution summary of the wafers average thickness values by sets of different product designs.

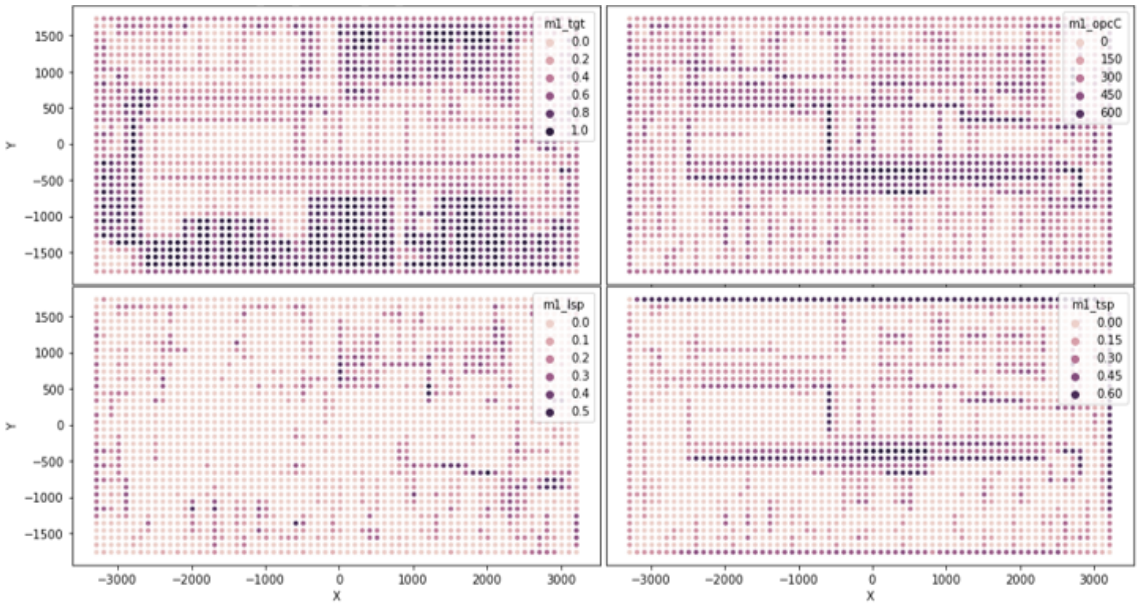


Figure 4.6: Example of a die layout of a specific design and its layout design characteristics extracted by Siemens.

the model may show poor performance while predicting for new (not seen during training) product designs.

A viable alternative could be the use of layout information on the design. The idea is to use the layout characteristics as numerical predictors instead of (or together with) the "product ID" information in the regression problems so that when a new design is introduced into the production line of the fab that has no correspondent in the training data set, then layout features (that would be available in the test data set) could be resorted and improve the accuracy of VM prediction.

As the matter of fact, each device has a specific layout, as depicted in Figure 4.6 and we can assume that the semiconductor has a reference system with a well-determined origin. Then, thickness represents a pointwise measure in a certain position (coordinate) of the chip belonging to a specific grid cell (die) on the wafer, depending on the spot. Such coordinates can be considered fixed across lots and wafers, but the specific coordinates of the semiconductor can be different for different spots (because for different spots, different coordinates of the semiconductor can be measured).

To replace important categorical information on the product with (as much as possible) equivalent numerical available information, we proposed to investigate layouts of several the most populated products (10 in total in this work) by dividing the entire region in a number of subregions (up to 4k), and for each subregion, some structural features have been extracted/estimated (see Figure 4.6). Once redundant Features have been discarded, 4 Features remain available (sampled in around up to 4k subregions), and mean values, variance and skewness are finally provided as tables. Since the up to 4k values are averaged all over the subregions, they represent global values, representative of the entire semiconductor, which is enough in case the target is an average thickness.

A different procedure is to consider for one particular spot thickness prediction. To that

spot a specific semiconductor in the cell grid corresponds, and the thickness depends on the position in the semiconductor (better, in a specific subregion of the semiconductor, whose area depends on δ). Therefore, in order to be consistent, Layout parameters should be given not globally, but for the subregion where Thickness measurement is taken. Operationally, the proposed approach can be described with the following steps:

1. Estimate δ making some assumptions on the region underlying δ and on the spatial accuracy of the thickness measurement;

2. Determine the coordinated of measurement on the semiconductor concerning a fixed reference system on the semiconductor that could depend on the layout;

3. Ont the layout grid, select the subregion of the semiconductor determined in 1) and 2);

4. Provide layout features (predictors for the developed VM framework) for that subregion.

The benefit of using the proposed continuous features describing design in the VM solver is shown experimentally further in Section 4.3.4.1.

4.3.2.4 Auxiliary Data

Recipe

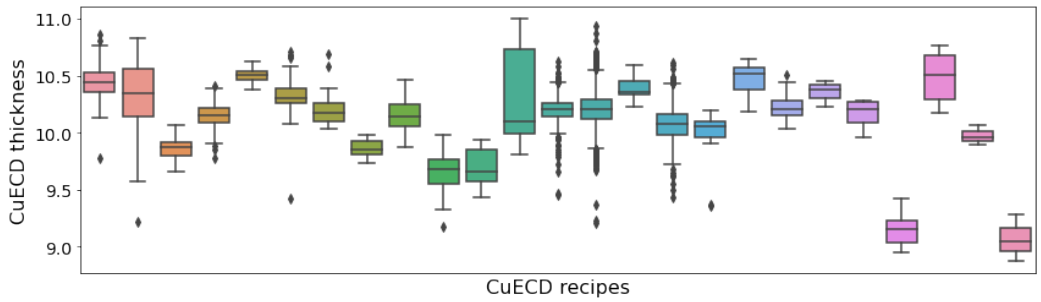
The recipe is a configuration to be set on the processing equipment and generally depends on the product design of the considered wafer. In this work, there exist 26 different recipes in the data set that are used to treat 146 different products. The number of recipes is normally smaller or equal then the number of products, as two distinct products may share common functionalities.

Figure 4.7a shows the variation of the thickness distribution characteristics depending on their recipe. Additionally, Figure 4.7b displays an evidence that wafers of different designs processed with a one common CuECD recipe may acquire different CuECD thickness outcome due to dependencies with the previously deposited layers that are different. This kind of influence is investigated in the next Section 4.4.

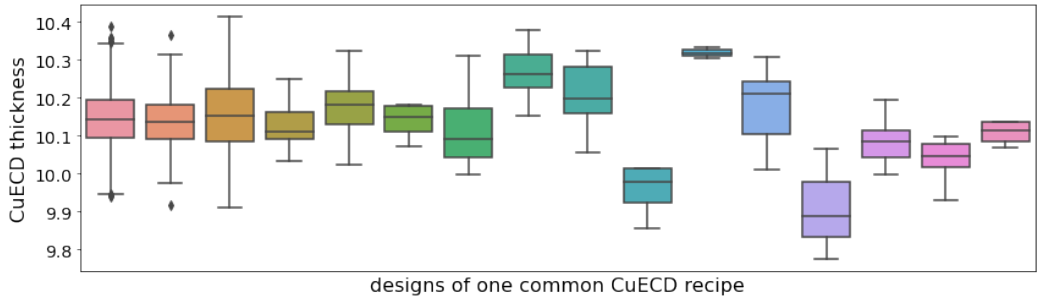
Since recipe and product design are somewhat related, "recipe ID" as a categorical feature (similarly to the "product ID") is an uninformative predictor or doesn't contribute to the generalization, since new recipes are always introduced with time in production. Thus, continuous predictors are considered, that represent the numbers to be set on the chamber in order to run the required process specifications.

Chamber

There is a bias in the metrology result depending on which chamber is used for the deposition, even if the chambers are thought to be equal and the usage of a specific one depends on the engineering process and its availability. The clear evidence of this phenomenon is depicted in Figure 4.8. The data available in this work was collected over 6 chambers of one CuECD

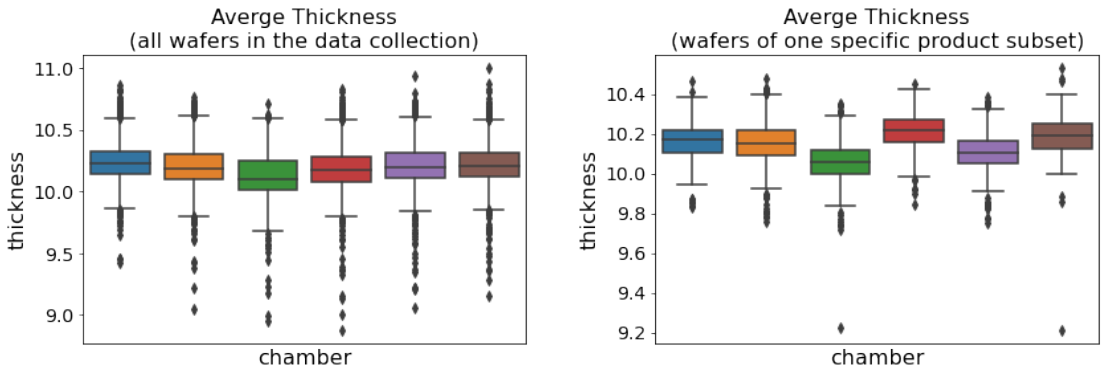


(a) The wafers average thickness statistics of sets formed of distinct CuECD process recipes.



(b) The wafers average thickness statistics of sets formed of distinct wafer designs but of one common CuECD process recipe.

Figure 4.7: Display of the distribution summary of the wafers average thickness values by sets of different groups (recipes, products).



(a) (b)

Figure 4.8: Display of the distribution summary of the set of the wafers average thickness values (a) of all the designs (b) of only one design, depending on the chamber they were processed at.

equipment. Then, we show in Figure 4.8 descriptive statistics of the sets of the wafers average thickness measurements collected from different chambers, and their difference is evident in both cases: when every wafer from every design in the data collection is used for statistics; when only one kind of product design is chosen for comparison.

Therefore, "chamber ID" is a necessary variable to be included in the set of predictors in order to explain the variability of the product outcome inside the equipment, and it is enough to keep it as categorical one (unlike "recipe ID" and "product ID"), since the machinery remain unchanged.

4.3.3 VM Solver

Mathematically the VM problem is defined as a regression one, and a lot of ML regression strategies discussed in Section 2.2.2 were adapted and proposed to develop the VM frameworks for CuECD application.

The predictive framework for CuECD VM problem in this thesis was experimented with the real SM data collection provided by STMicroelectronics, which consists of the ESD records of several chambers collected during 2019-2020 years. First, the given data was preprocessed accordingly to the methods described in the previous Section 4.3.2. Moreover, this case study operates with over a hundred product labels, tens of distinct recipes and several different chambers, all are denoted in the dataset by unique names. Then, categorical feature embedding methods are used to transform the names into numerical labels, for example, the One-Hot-Encoding approach or encoding with a method that associates each category with a unique discrete number (randomly or using some defined process, for example, bigger numbers are assigned to more populated categories).

As far as regression is concerned, in models where only uncorrelated features should enter, only the uncorrelated features are considered, together with one representative feature from each cluster of highly correlated ones. The one is chosen according to the best predictive power based on univariate Linear Regression (LR), eventually with "chamber ID", "Recipe ID", and "product ID" as control variables.

4.3.3.1 Experimental Setup

In the entire project, it is assumed that data from 2019 are used to train regression. In addition, the models are also tested on data from 2020 (Test data set), that are close to industrial conditions since far away enough from the training data set and with new devices. Most models require an estimate of hyperparameters; this is accomplished by splitting the 2019 data into two disjoint data sets, a Training one and a Validation one. This methodology, consolidated in scientific research, avoids overfitting due to estimating error indicators on data sets that have been used in setting a model and tuning its parameters.

The number of available samples (wafers) for the 2019 (Training + Validation) and 2020 (Test) data set includes 10000 ESD wafers. One of the most consolidated methods for splitting a data set into Training and Validation subsets is K-fold Cross Validation (CV), in particular a 10-fold CV with K=10. Essentially the entire data set (2019) is split into K=10 disjoint groups approximately of the same size. They generate 10 different sets of Validation and Training data; in each one, the Validation set is given by one of the 10 groups with 10% of data, and the Training set is made of the remaining 90% of data. In this way, Training and Validation data are disjoint for each group. To estimate an error indicator, the model is run 10 times, each time on a couple of Training-Validation, where the Training data set is used to train the model (and hyperparameters, eventually), and error is estimated

on the Test data set. The final estimate of the error indicator is obtained by averaging the error indicators of the 10 run.

Despite its simplicity, the random generation of K-fold groups needs a deeper understanding in some circumstances. In fact, nothing was said about how the K groups are randomly selected from the original samples. However, a plain random selection can cause troubles with the representativeness of other variables both in the Training and Validation groups. This is exactly the case of the SM data collections, namely with the presence of the variables "recipe ID" or "product ID". It can happen that in one of the Validation data sets randomly chosen, there exist "recipe ID"s or "product ID"s that are not represented in the corresponding Training data set. In this case estimate for such a "recipe ID"s or "product ID"s on the Validation data set is not possible, and the resulting error indicator is biased. Another problem arises with "recipe ID"s or "product ID"s that are less populated, let us say less than K samples. In this case, it is sure that for some CV groups, the corresponding "recipe ID" or "product ID" will be missing, again producing a bias in the error indicator. To overcome these problems, the following actions were taken:

- Randomly select CV groups stratifying by "recipe ID" or "product ID". In other words, random selection is not made on the whole sample, by "recipe ID" by "recipe ID" (or "product ID" by "product ID"), so that it is sure that all "recipe ID"s (or "product ID"s) are present both in the Training and Validation data sets
- In the case of less populated "recipe ID"s (or "product ID"s; less than K), the sample is artificially increased up to K by random selection of K samples with repetitions. Some bias is still introduced in the error indicator, however, experiments show that it is much better controlled. As an alternative, such less populated "recipe ID"s or "product ID"s can be removed from the analysis.

Comparison of different VM methods in this work is accomplished by estimating the coefficient of determination R^2. It is a statistical measure in the range [0,1], which shows a percentage of the dependent variable variation that a linear model explains:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y^i_{real} - y^i_{predicted})^2}{\sum_{i=1}^{N}(y^i_{real} - \frac{1}{N}\sum_{i=1}^{N} y^i_{real})^2}.$$

Higher R^2 values represent smaller differences between the observed data and the fitted values.

4.3.4 Experimental Results

In this Section, we present a comparison of the methods to predict CuECD VM. For each case estimate of R^2 will be provided on the train data set; for these experiments, the training data set is rather improperly composed of the entire year 2019, also used for validation. Then an estimate of R^2 is shown on the test data set composed of the entire year 2020. In this case, the test data set is totally disjoint from the training data set.

First, we investigate the results with some Basic models in Table 4.2, where Basic is intended for models that do not involve process information, but only ancillary variables (chamber, recipe, product).

Table 4.2: Predictability of basic models: regression involving only ancillary variables (recipe, product and chamber information without any ESD). The average value computed on the Training data set stratified by "chamber ID" and "product ID".

Model	Validation score (2019)	Test score (2020)
"recipe ID"	0.633	0.309
"recipe ID" + "chamber ID"	0.641	0.325
"product ID"	0.756	0.484
"product ID" + "chamber ID"	0.764	0.598

It is possible to observe that the simple Basic models are immediately useful, since they give some indication of the R^2 that it is possible to expect from more accurate and elaborated models, and also because they easily compare different models based on different involved ancillary variables. Not shown in the Table, models with interactions between all recipes, products, and chambers have also been run. Despite the slight increase of R^2 in the Training data set with respect to models without interaction, sometimes a decrease of R^2 in the Test data set is observed. This is probably justified by the higher number of predictors, which induces some overfit of the model. Analogously, a model with both recipe and product (without interaction) does not improve R^2 obtained with only Product as a predictor. Results are not shown for the sake of brevity.

Table 4.3: Predictability of different models: regression involving both ancillary variables (recipe, product, and chamber information) and ESD. The average value computed on the Training data set stratified by "chamber ID" and "product ID".

Model	Validation score (2019)	Test score (2020)
GBDT	**0.895**	**0.652**
RF	0.874	0.613
SVR	0.786	0.272
Full Stepwise Regression	0.700	0.478
Penalisation	0.769	0.520
Group Penalization	0.764	0.484
SLR	0.794	0.339
ANN	0.863	0.614

Next, 8 different Regression methodologies have been compared and reported in Table 4.3, and the one yielding top performance is GBDT, with a rate as high as 91% of explained variance on the 2019 Validation data set, with RF and ANN showing very close performance. Noteworthy, simple models were already able to give decent results; they are easier in the interpretation of the model, and allow one to get a good insight into the system.

It is to be mentioned that in all models categorical variables "recipe ID" and or "product ID" (or corresponding dummy variables when used) have the highest influence in predicting thickness. This explains why simple models like Basic or linear reach comparably high performance on the Validation data set and on the Test one.

Table 4.4: Predictability of GBDT model depending if product descriptive features are included or not in the set of input predictors.

Test Product	Without Design Features		With Design Features	
	R^2	MSE	R^2	MSE
$prod56$	0.112 ($\pm$ 0.086)	0.019 ($\pm$ 0.002)	**0.464 ($\pm$ 0.054)**	**0.011 ($\pm$ 0.001)**
$prod81$	0.245 ($\pm$ 0.094)	0.017 ($\pm$ 0.004)	**0.349 ($\pm$ 0.073)**	**0.014 ($\pm$ 0.003)**
$prod157$	-5.105 ($\pm$ 1.223)	0.015 ($\pm$ 0.014)	**-3.447 ($\pm$ 0.808)**	**0.011 ($\pm$ 0.007)**
$prod325$	-3.014 ($\pm$ 0.389)	0.176 ($\pm$ 0.017)	**-2.013 ($\pm$ 0.214)**	**0.135 ($\pm$ 0.009)**
$prod1221$	**-2.126 ($\pm$ 0.216)**	**0.074 ($\pm$ 0.004)**	-3.024 ($\pm$ 0.238)	0.083 ($\pm$ 0.005)
$prod1306$	-0.905 ($\pm$ 0.216)	0.05 ($\pm$ 0.004)	**-0.649 ($\pm$ 0.222)**	**0.039 ($\pm$ 0.003)**
$prod1312$	0.12 ($\pm$ 0.084)	0.022 ($\pm$ 0.008)	**0.135 ($\pm$ 0.061)**	**0.022 ($\pm$ 0.007)**
$prod1321$	0.298 ($\pm$ 0.084)	0.013 ($\pm$ 0.002)	**0.362 ($\pm$ 0.065)**	**0.013 ($\pm$ 0.001)**
$prod1336$	-0.186 ($\pm$ 0.151)	0.045 ($\pm$ 0.005)	**0.12 ($\pm$ 0.098)**	**0.031 ($\pm$ 0.003)**
$prod1337$	-0.988 ($\pm$ 0.273)	0.036 ($\pm$ 0.003)	**0.298 ($\pm$ 0.093)**	**0.013 ($\pm$ 0.002)**

As for the predictability of the 2020 data set, never seen by the Training data set, presented in order to test the real use of the tools in an industrial environment. Despite the presence of new "recipe ID"s and "product ID"s, not included in the learning 2019 data set, and some variability in the ESD, predictability is still acceptable, as high as 66% by Gradient Boosting.

4.3.4.1 Design Features Importance

Here, we investigated the possibility to boost the performance of the VM solver by introducing additional predictors of design characteristics explained in Section 4.3.2. We preselected 10 of the most populated product groups in the given data set (extracting proposed features for all of the 146 designs available in the full data fn the project would be very expensive and therefore only a few were considered for the assumption proof) and for them, we extracted 4 continuous descriptional features.

Our main focus was to check if we can enhance the generalization (accuracy rate evaluated with 2020, which is affected mostly by the presence of new products) with the use of the product predictors, and in particular, if we can improve the accuracy of metrology estimation for new product designs (not seen during training). Accordingly, a "leave one design out" experiment was proposed, where 10 different models were trained with the wafers data set composed of only 9 different products while leaving the last one for testing. Then, each of the models was trained with a different predictor, once including the extracted design characteristics and without them. The results were compared and reported in Table 4.4. The improvement is obvious except for the one product ($prod1221$).

In order to find a possible explanation why for some of the products the improvement is observed and why for some not, in parallel, we analyzed "closeness" of the considered designs based on the distance between them in the reduced dimension space of the extracted design charachteristics (see Figure 4.9). From this experiment we learned that the boost in accuracy of predicted metrology values for a new product is more probable in case the new design is "not far" from at least one product already presented in the training set. The

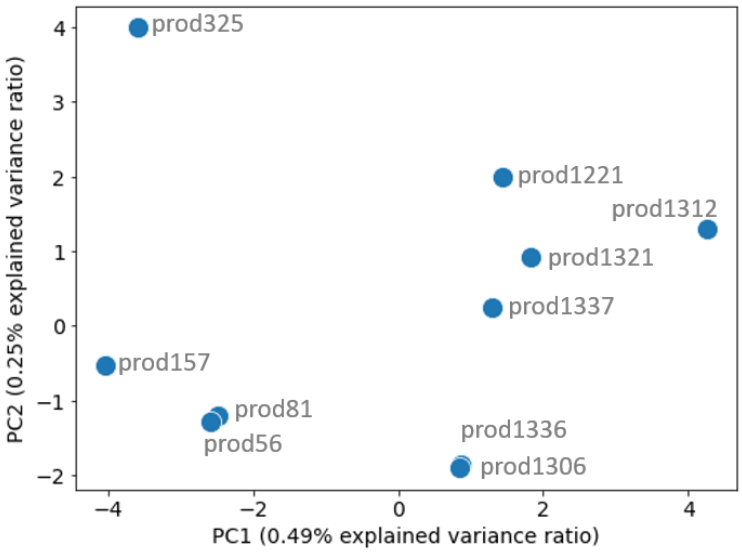


Figure 4.9: PCA representation of design features of 10 different products.

evidence for that is given if the Figure 4.9 is followed with the Table 4.4.

4.4 Virtual Cross Metrology

Now, to improve the predictability, we investigate the assumption that the variability of the outcome observed across final measurements can be explained not only by the product variability but also can be derived by analyzing the full history of a wafer. Accordingly, it may better characterize manufacturing operations and help identifying root causes effects outside the environment of the considered process step. As the process layers on a wafer are sequentially manufactured (see Figure 4.10), one can use the prior history to gain insight in the root causes affecting the final target.

In this Section, we introduce a Virtual Cross Metrology (VCM) approach that provides an objective quantification of the benefits of different measurements, detection, and quantification of dependencies between different process steps, and the opportunity to make processes more predictable and productive. In other words, the VCM system, presented in Figure 4.10, has to be a set of models that get generated, so that they are able to complete the full stack of all of the different processes metrology steps. The difference between those models compared to the individual processes step VM is that they carry around metrology information from the prior processes steps with the goal to improve the accuracy and interpretability by studying what is the relative contribution of prior information to the target estimation.

4.4.1 Experimental Results

The work presented in this Section was carried out on data collected in the production lines of IMEC enterprise. These data are mainly guided by the experience of process engineers

in assessing the condition of the wafers during the manufacturing process. Overall, for a collection of 20 wafers of one specific product, we were given data from 12 different consecutive process steps with their corresponding metrology sampling for most of the fields (that are in total 140 on one wafer). For each process-metrology pair two different models were trained: the first one refers to the VM approach introduced in the previous Section 4.3; and the second one refers to the VCM approach when the input set for the model is extended with respect to predictors and targets of all the prior process-metrology pairs to the considered one.

The experimental setup repeats the one introduced in the previous Section 4.3.4, while the predictive model was chosen to be GBDT as it is the one that provided the best predictive performance for the VM use case (see Section 4.3.4). Importantly, it has to be noted that not all wafers have their metrology probes sampled at every process step, then the missing values are imputed by the predictive models created at the corresponding process step in order to be used further for the VCM methodology.

In Table 4.5 and Figure 4.11 we show the accuracy results for the metrology estimation for the final 12-th process of the considered data set for both VM and VCM methodologies. As the considered process finishes the manufacturing of the IC, its metrology probes are the final electrical testing results to validate the proper functioning of the produced product. In total, we conducted the experiment for 14 different electrical tests predictions, and in most of the cases the VCM approach outperforms the VM framework as high as 6% on the average improvement of R^2.

However, for some of the targets, like $Metrology08$, $Metrology10$, and $Metrology12$, it is observed that the VCM approach is worst than the VM. This is probably justified by the higher number of predictors used by the VCM model that may cause some overfit of the model under low sample conditions (20 wafers only in the considered data collection). Indeed, the results are presented for the last in the production chain metrology values, then

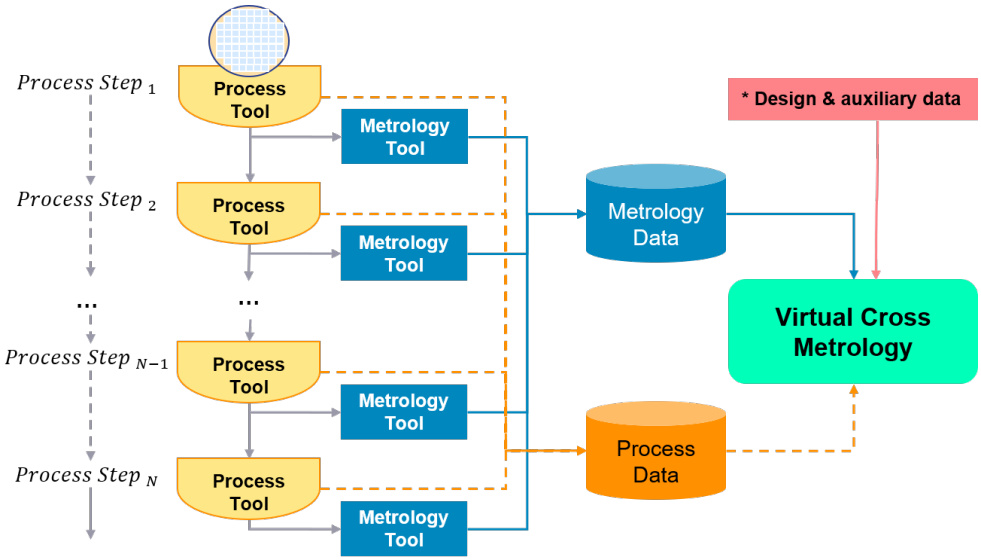
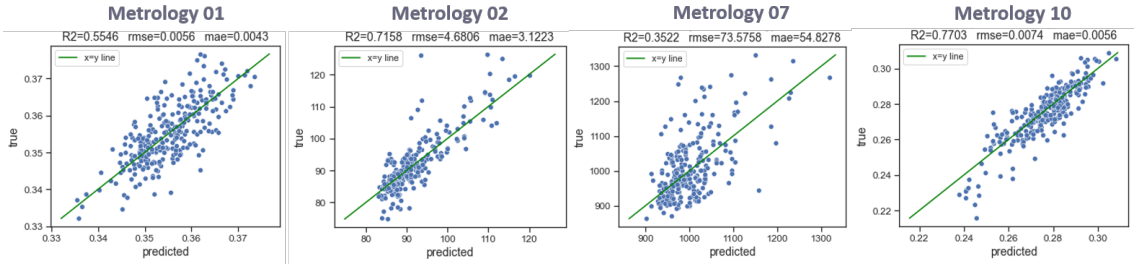


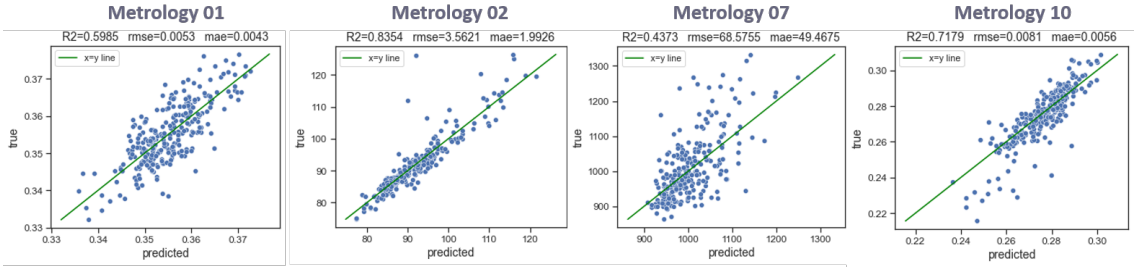
Figure 4.10: Schematic representation of the full chain of processes applied to one wafer to fabricate ICs on it.

Table 4.5: Predictability comparison with the VM and VCM approaches.

Electrical test	VM		VCM	
	R^2	RMSE	R^2	RMSE
$Metrology01$	0.5333	0.0057	**0.5808**	**0.0053**
$Metrology02$	0.7057	4.4609	**0.8641**	**2.9958**
$Metrology03$	0.4863	8.1513	**0.6122**	**7.0774**
$Metrology04$	0.6686	96.6611	**0.6966**	**92.5615**
$Metrology05$	0.7037	88.6968	**0.7341**	**83.9846**
$Metrology06$	0.2835	73.4609	**0.3661**	**69.0785**
$Metrology07$	0.3005	70.9557	**0.3698**	**67.3049**
$Metrology08$	**0.3369**	**199.2691**	0.3343	199.6938
$Metrology09$	0.2979	147.4196	**0.3013**	**147.127**
$Metrology10$	**0.7174**	**0.0073**	0.6691	0.0078
$Metrology11$	0.1291	0.6709	**0.1415**	**0.6656**
$Metrology12$	**0.2169**	**75.0395**	0.1068	80.1292
$Metrology13$	0.7533	7.7545	**0.7841**	**7.2791**
$Metrology14$	0.7134	6.1605	**0.7681**	**5.5462**



(a) Display of the predictability of different metrology electrical tests results with VM approach.



(b) Display of the predictability of different metrology electrical tests results with VCM approach.

Figure 4.11: Predictability comparison with the VM and VCM approaches.

in the case of VCM approach the predictors and targets from all of the prior processes are included as an input, which in our case means that the size of predictors set grows in 12 times on average.

As a result, the VCM proves the benefit of leveraging the wealth of information collected during the full cycle of the manufacturing procedure to fully characterize the process outcome, while an overfitting is a possible limitation of the proposed approach in case of low sampling.

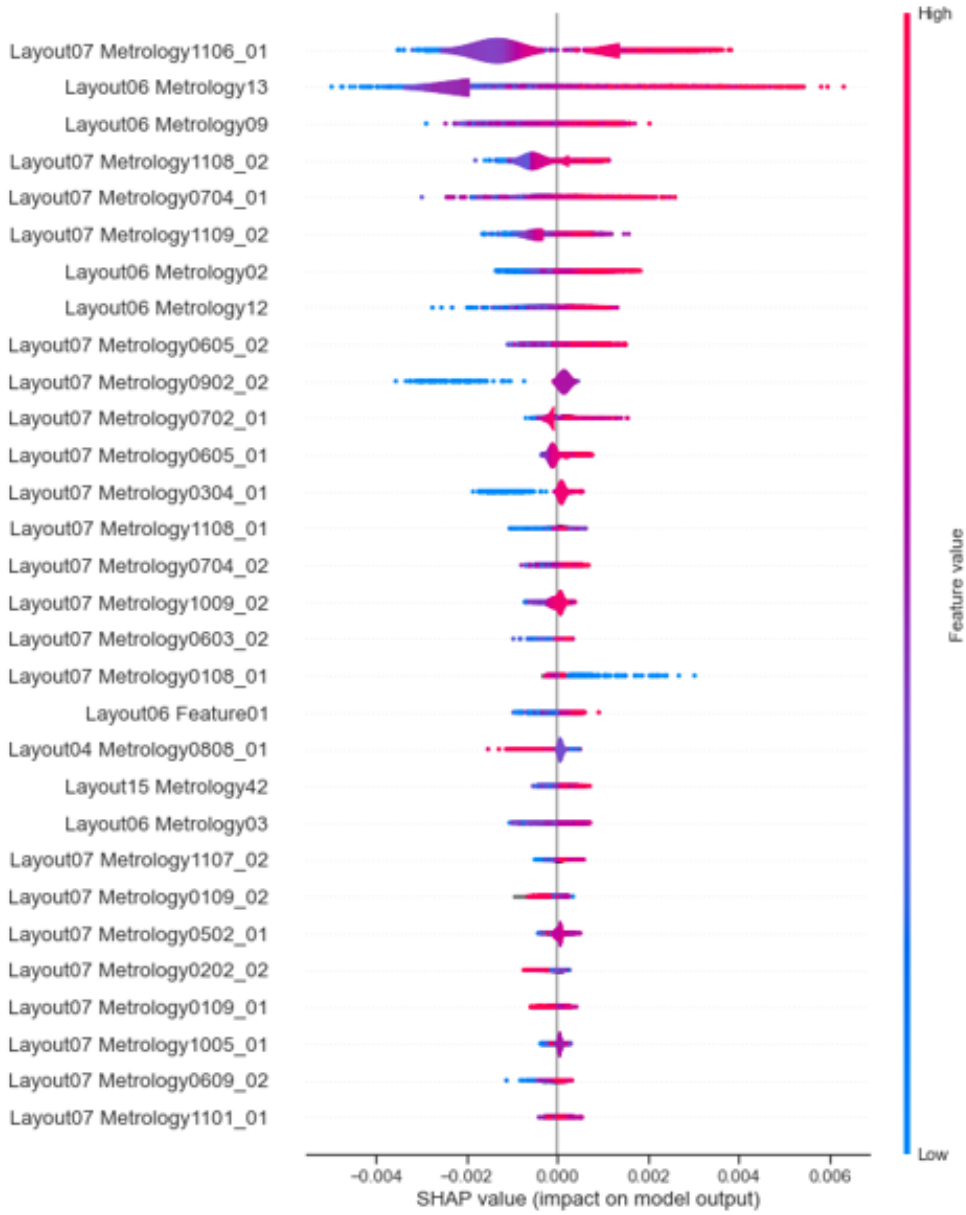


Figure 4.12: Display of the VCM model SHAP analysis for the $Metrology01$.

After we have generated such models we can identify the value of prior information and what are the characteristics or what are parameters that lead to low or high thickness conditions. ML techniques, like SHAP analysis, allow us to determine what are the main contributors among the full set of given predictors on the model output. An example of such analysis is given in the left plot in Figure 4.12, where the top-ranked predictors are depicted in the ordered manner form the most to the least important ones.

Then, identifying what are the top features impacting the model outcome for each of the targets at each of the process steps allows for determining how the processes are dependent on each other. For example, Table 4.6 shows the processes in which features are present in the top 30 predictors influencing the prediction of each of the electrical test predictions

Table 4.6: Display of the results of the SHAP analysis conducted per each metrology VCM model in order to discover the processes steps that contribute the most to explaining the electrical test data.

ETest	Process										
	$P01$	$P02$	$P03$	$P04$	$P05$	$P06$	$P07$	$P08$	$P09$	$P10$	$P11$
$Metrology01$			✓		✓	✓					✓
$Metrology02$			✓		✓	✓		✓			
$Metrology03$			✓		✓	✓					
$Metrology04$			✓		✓	✓		✓	✓	✓	✓
$Metrology05$			✓	✓	✓	✓		✓	✓		✓
$Metrology06$			✓	✓	✓	✓	✓		✓	✓	✓
$Metrology07$			✓	✓	✓	✓	✓				
$Metrology08$			✓		✓	✓	✓	✓	✓	✓	✓
$Metrology09$			✓		✓	✓		✓	✓	✓	
$Metrology10$			✓		✓	✓			✓	✓	✓
$Metrology11$	✓		✓		✓	✓		✓		✓	✓
$Metrology12$			✓	✓	✓	✓		✓		✓	
$Metrology13$			✓	✓	✓	✓		✓		✓	✓
$Metrology14$			✓		✓	✓				✓	

by the VCM model. Why is it important? Because it allows identifying the processes like $Process02$ from the Table 4.6 that don't impart the estimation of the final target, therefore metrology may not be measured at the process step in order to increase productivity unless they are important in any other control task.

4.4.2 Discussion

At the beginning of the MADEin4 project, in the fab protocol it was mandatory to measure at least six wafers for each lot in order to keep it under control chambers performance. The developed CuECD VM framework proposes a practical ML solutions allowing to the addition of virtual measurements to all wafers belonging to the same lot. Moreover, we proposed a novel approach to introduce design characteristics that were proven in practice to enhance the accuracy of the estimated values.

The framework was deployed to the STMicroelectronics infrastructure and was tested in a demo environment to reduce the sampling frequency in a way that real metrology automatically activated only when a critical issue emerges in the VM analysis. The results demonstrate the efficiency of presented practical approaches to catch in advance the upcoming issue and consequently to increase the operational efficiency (OE), defined in Equation 4.2, more than 20% with pre-MADEin4 value at the beginning of the project.

$$OE = \frac{Processing\ time}{Uptime} \tag{4.2}$$

Increasing the OE it is possible to preserve potential process drift on devices production and improve the products' yield.

Additionally, we proposed to expand the scope of traditional process modeling in the SM by cross-process analysis, called VCM. The experiments conducted with the real fab data given by the IMEC enterprise show the average increase by the VCM approach as high as 6% reaching its maximum up to 16%.

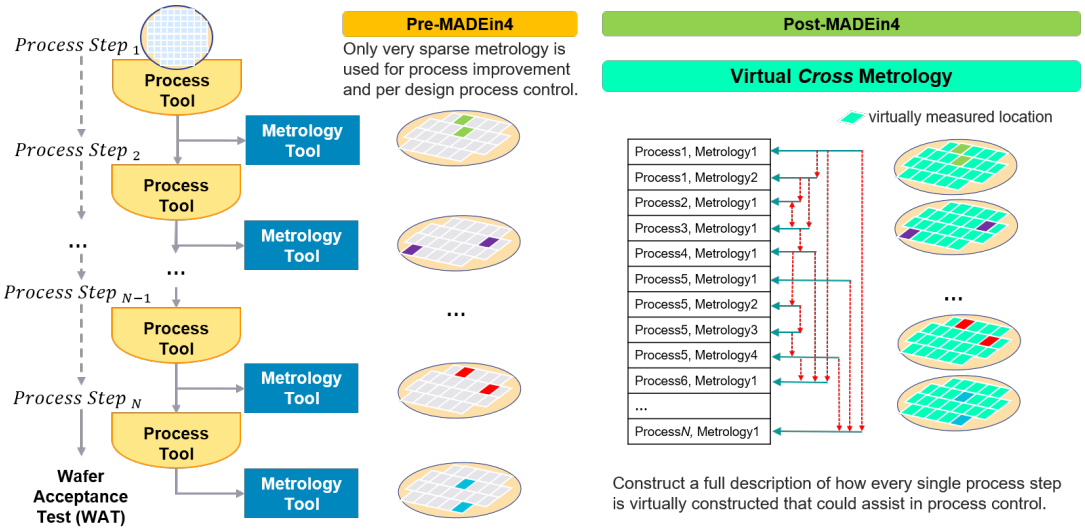


Figure 4.13: Pre-MADEin4 shows the sparsity of the current metrology sampling during the manufacturing in order to conduct process and product control related tasks. Post-MADEin4 indicates the results of using design metrology and process information of previous processes steps to generate the full view of the wafer history.

As a result, in this Chapter, we showed that by having the same number of physical measurements we can fill in the "blanks" by generating intermediate (for every process step) VM models to improve the coverage of the metrology sampling. We showed that the benefit of design features aids in the intermediate probes models, which in turn reduces the uncertainty of the predictions. However, the major benefit is also found in the application of the cross metrology technique which enables the full stack characterization, when we can create models that now utilize prior real and virtual measurements to improve accuracy. Furthermore, the developed IT architectures can be reused for the next developments in this field, which can be:

- An improvement of the VM method for handling several recipes or products through gradient boosting endowed with mixed effect models;
- An investigation of a transfer learning approaches to enhance the generalization through making more accurate predictions when the new recipes and products appear;
- Better use of maintenance (both PvM and PdM) data that affect time series of ESD for better identification and characterisation of the process drift.

Chapter 5

Generative Adversarial Networks for Multi-view Learning with Missing Views

This chapter is based on the paper [DA20].

5.1 Motivation

During the last years, the number of ML tasks that employ data from several different sources has increased considerably. Accordingly, a direction, called multi-view learning, was defined to propose different methods that can effectively learn from diverse sets of features that define very the same object; and many advances have been made on both theoretic and algorithmic sides in this direction. The three main families of techniques for multi-view learning are: 1) CCA that finds pairs of highly correlated subspaces between the views that is used for mapping the data before training, or integrated into the learning objective [6, 28]; 2) MKL that considers one kernel per view and different approaches have been proposed for their learning [7]; and 3) co-regularization that tend to minimize the disagreement between the single-view classifiers over their outputs by adding a regularization term to the objective function [60, 71].

Overall, well designed multi-view learning strategy has better generalization ability than single-view learning [74]. However, all mentioned above techniques assume that the views of samples are complete and available during training and testing. But in practice observations often have missing data, which raises a multi-view learning problem in the case where some observations may have missing views without there being an external resource to complete them. This is a typical situation in many applications where: 1) one/some of the different sources that generate the views are not available at a time (like all Wikipedia pages contain text information, while images content is more scarce); 2) different sources generate different views of samples unevenly (like equipment sensors in the semiconductor manufacturing fab). Moreover, it can be expensive to collect data from all available sources, which is mostly the case for many industrial problems, and therefore companies should compromise between predictive accuracy to its cost.

Previous works supposed the existence of view-generating functions to complete the missing views before deploying a learning strategy [3]. However, the performance of the

global multi-view approach is then biased by the performance quality of the generating functions, which generally require external resources to be set. The challenge is hence to learn an efficient model from the multiple views of training data without relying on an extrinsic approach to generate altered views for samples that have missing ones.

In this direction, GANs provide a propitious and broad approach with a high ability to seize the underlying distribution of the data and create new samples [32]. These models have been mostly applied to image analysis, and major advances have been made on generating realistic images with low variability [23, 51, 56]. In the direction of learning from a different view, some works included an inverse mapping from the input to the latent representation, mostly referred to as BiGANs, and showed the usefulness of the learned feature representation for auxiliary discriminant problems [24, 26]. This idea paved the way for the design of efficient approaches for generating coherent synthetic views of an input image [69, 46, 19]. For instance, GANs are successfully applied for image domain transfer or missing pixels imputation, which made us to assume that they also can serve for missing views imputation and be adapted for joint learning.

One common example of multi-view tasks with missing views, that we consider in this chapter to study the challenge of joint learning of missing view imputation and target prediction, is a multilingual text classification where documents are available in several languages and share the same set of classes while some are just written in one or more, but not all languages.

5.2 Contributions

We propose a cGAN-based model, called Cond^{2}GAN, that employs two generators and a common discriminator to solve multi-view learning problems where observations have two views, but one of them may be missing for some of the training samples.

Particularly, we consider a bilingual text classification problem, where majority of training documents are written in only one language; and the proposed model learns the representation of missing versions of bilingual documents jointly with the association to their respective classes. As mentioned, Cond^{2}GAN is composed of two generators G_1 and G_2 and one discriminator D formulated as a tripartite game. For a given document with a missing version in one language, the corresponding generator induces the latter conditionally on the observed one. The training of the generators is carried out by minimizing a regularized version of the cross-entropy measure proposed for multi-class classification with GANs [61] in a way to force the models to generate views such that the completed bilingual documents will have high class assignments. At the same time, the discriminator learns the association between documents and their classes and distinguishes between observations that have their both views and those that got a completed view by one of the generators. This is achieved by minimizing an aggregated cross-entropy measure in a way to force the discriminator to be certain of the class of observations with their complete views and uncertain of the class of documents for which one of the versions was completed. The regularization term in the objectives of generators is derived from an adapted feature matching technique [59], which is an effective way of preventing from situations where the models become unstable; and which leads to fast convergence.

We demonstrate that generated views allow achieving state-of-the-art results on a subset of Reuters RCV1/RCV2 collections compared to multi-view approaches that rely on Machine Translation (MT) for translating documents into languages in which their versions do not exist; before training the models. Importantly, we exhibit qualitatively that generated documents have meaningful translated words bearing similar ideas compared to the original ones; and that, without employing any large external parallel corpora to learn the translations as it would be the case if MT were used. More precisely, this work is the first to :

- Propose a new tripartite GAN model that makes class prediction along with the generation of high-quality document representations in different input spaces in the case where the corresponding versions are not observed (Section 5.4);
- Achieve state-of-the-art performance compared to multi-view approaches that rely on external view generating functions on multilingual document classification; and which is another challenging application than image analysis which is the domain of choice for the design of new GAN models (Section 5.5);
- Demonstrate the value of the generated views within our approach compared to when they are generated using MT (Section 5.5);
- Showcase the use of the proposed tripartite GAN model with the image data as well as semiconductor data collections.

5.3 Problem Setting

We consider the multi-label bilingual text classification problem, where documents are represented as feature vectors using a TFIDF-based weighting scheme. Accordingly, one document in a data collection, considering one of the languages $l \in \{1, 2\}$, is a feature vector x^l - being a bag of words representation in a corresponding language vocabulary of size d_l. Then, a bilingual document is defined as a sequence $\mathbf{x} = (x^1, x^2) \in \mathcal{X}$ that belongs to one and only one out of K different classes. The class membership indicator vector $\mathbf{y} = (y_k)_{1 \leq k \leq K} \in \mathcal{Y} = \{0, 1\}^K$, of each bilingual document, has all its components equal to 0 except the one that indicates the class associated with the example which is equal to one. Since we consider a specific setting when a majority of documents are written in only one language, we suppose that $\mathcal{X} = (\mathcal{X}_1 \cup \{\perp\}) \times (\mathcal{X}_2 \cup \{\perp\})$, where $x^l = \perp$ means that the l-th language representation is missing (the corresponding view is not observed). Hence, each observed view $x^l \in \mathbf{x}$ is such that $x^l \neq \perp$ and it provides a representation of $\mathbf{x}$ in a corresponding input space $\mathcal{X}_l \subseteq \mathbb{R}^{d_l}$.

We assume that each example $(\mathbf{x}, \mathbf{y})$ is identically and independently distributed (i.i.d.) according to a fixed yet unknown distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, and that at least one of its views is observed. Furthermore, following the conclusions of the co-training study [11], our work is based on a next main assumption :

Assumption 1 ([11])**.** *Observed views are not completely correlated, and are equally informative.*

In our framework, we suppose to have access to a training set $\mathcal{S} = \{(\mathbf{x}_i, \mathbf{y}_i); i \in \{1, \ldots, m\}\} = \mathcal{S}_F \sqcup \mathcal{S}_1 \sqcup \mathcal{S}_2$ of size m drawn i.i.d. according to $\mathcal{D}$ (Figure 5.1), where

- $\mathcal{S}_F = \{((x_i^1, x_i^2), \mathbf{y}_i) \mid i \in \{1, \ldots, m_F\}\}$ denotes the subset of training samples with their both complete views;
- $\mathcal{S}_1 = \{((x_i^1, \perp), \mathbf{y}_i) \mid i \in \{1, \ldots, m_1\}\}$ is the subset of training samples with their only first view available
- $\mathcal{S}_2 = \{((\perp, x_i^2), \mathbf{y}_i) \mid i \in \{1, \ldots, m_2\}\}$ is the subset of training samples with their only second view available;
- and $m = m_F + m_1 + m_2$.

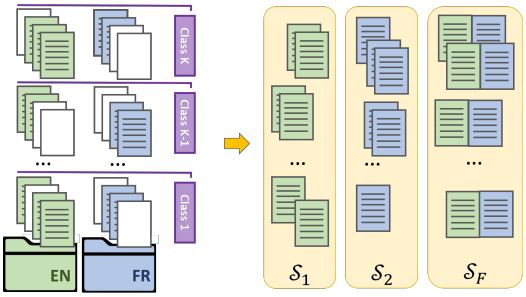


Figure 5.1: Graphical explanation how data is divided into $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_F$ subsets.

Overall, it is possible to fit such classification problem using existing techniques. For example, by learning single view classifiers independently on the examples of $\mathcal{S} \sqcup \mathcal{S}_1$ and $\mathcal{S} \sqcup \mathcal{S}_2$ for class label estimation based on each of the views independently. Then, to make predictions, one can then combine the outputs of the classifiers (or learn a fusion of labels) [68] if both views of a test example are observed, or otherwise, use one of the outputs corresponding to the observed view. Another solution is to apply multi-view approaches over the training samples of $\mathcal{S}_F$ only; or over the whole training set $\mathcal{S}$ by completing the views of examples in $\mathcal{S}_1$ and $\mathcal{S}_2$ before using external view generation functions (like MT approach considering working with textual data). Accordingly, those existing techniques will serve for us to compare our proposed model within the experimental Section .

5.4 Cond^{2}GAN

As an alternative, the learning objective of our proposed approach is to generate the missing views of examples in $\mathcal{S}_1$ and $\mathcal{S}_2$, jointly with the learning of the association function between the multi-view samples (with all their views complete or completed) and their classes. The proposed model consists of three neural networks that are trained using an objective implementing a three players game between a discriminator, D, and two generators, G_1 and G_2. The game that these models play is depicted in Figure 5.2 and it can be summarized as follows. At each step, if an observation is chosen with a missing view, the corresponding generator – G_1 (respectively G_2) if the first (respectively second) view is missing – produces the view from random noise conditionally on the observed view in a way to fool the discriminator. On the other hand, the discriminator takes as input an observation with both of its views complete or completed and, classifies it if the views are initially observed or tell if a view was produced by one of the generators.

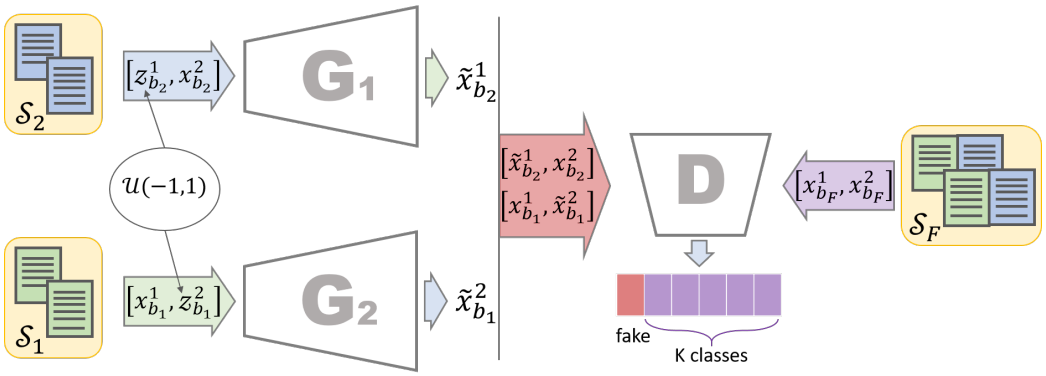


Figure 5.2: A visual representation of the proposed GAN model composed of three neural networks; a discriminator D and two generators G_1 and G_2. The missing view of an observation is completed by the corresponding generator conditionally on its observed view. The discriminator is trained to recognize between observations having their views completed and those with complete initial views as well as their classes.

5.4.1 Generators

Formally, both generators G_1 and G_2 take as input; samples from the training subsets $\mathcal{S}_2$ and $\mathcal{S}_1$ respectively; as well as random noise drawn from a uniform distribution defined over the input space of the missing view and produce the corresponding pseudo-view, which is missing:

- $G_1(z^1, x^2) = \tilde{x}^1$,
- $G_2(x^1, z^2) = \tilde{x}^2$,

where z^1 and z^2 being an input random noise vectors to the generators G_1 and G_2 respectively. These models are learned in a way to replicate the conditional distributions $p(x^1|x^2, z^1)$ and $p(x^2|x^1, z^2)$; and inherently define two probability distributions, denoted respectively by p_{G_1} and p_{G_2}, as the distribution of samples if both views where observed i.e. $(\tilde{x}^1, x^2) \sim p_{G_1}(x^1, x^2)$, $(x^1, \tilde{x}^2) \sim p_{G_2}(x^1, x^2)$.

5.4.2 Discriminator

On the other hand, the discriminator takes as input a training sample; either from the set $\mathcal{S}_F$, or from one of the training subsets $\mathcal{S}_1$ or $\mathcal{S}_2$ where the missing view of the example is generated by one of the generators accordingly. The task of D is then to recognize observations from $\mathcal{S}_1$ and $\mathcal{S}_2$ that have completed views by G_1 and G_2 and to classify examples from $\mathcal{S}_F$ to their true classes. To achieve this goal we add a fake class, $K+1$, to the set of classes, $\mathcal{Y}$, corresponding to samples that have one of their views generated by G_1 or G_2. The dimension of the discriminator's output is hence set to $K+1$ which by applying softmax is supposed to estimate the posterior probability of classes for each multiview observation (with complete or completed views) given in input. For an observation $\mathbf{x} \in \mathcal{X}$, we use $D_{K+1}(\mathbf{x}) = p_D(y = K+1|\mathbf{x})$ to estimate the probability that one of its views is

generated by G_1 or G_2. As the task of the generators is to produce good quality views such that the observation with the completed view will be assigned to its true class with high probability, we follow [59] by supplying the discriminator to not get fooled easily as stated in the following assumption :

Assumption 2 ([59])**.** *An observation* $\mathbf{x}$ *has one of its views generated by* G_1 *or* G_2*; if and only if* $D_{K+1}(\mathbf{x}) > \sum_{k=1}^{K} D_k(\mathbf{x})$.

In the case where; $D_{K+1}(\mathbf{x}) \leq \sum_{k=1}^{K} D_k(\mathbf{x})$ the observation $\mathbf{x}$ is supposed to have its both views observed and it is affected to one of the classes following the rule:

$$\max_{k=\{1,\dots,K\}} D_k(\mathbf{x})$$

.

5.4.3 The Tripartite Game

The overall learning objective of Cond^{2}GAN is to train the generators to produce realistic views indistinguishable with the real ones, while the discriminator is trained to classify multi-view observations having their complete views and to identify view-generated samples. If we denote by p_{real} the marginal distribution of multi-view observations with their both views observed (i.e. $(x^1, x^2) = p_{real}(x^1, x^2)$); the above procedure resumes to the following discriminator objective function $V_D(D, G_1, G_2)$:

$$\begin{aligned} \max_D V_D(D, G_1, G_2) = \; & \mathbb{E}_{(x^1,x^2,y)\sim\mathcal{S}_F}\left[\log p_D(y|x^1, x^2, y < K+1)\right] \\ + \; & \frac{1}{2}\mathbb{E}_{(\tilde{x}^1,x^2)\sim p_{G_1}}\left[\log p_D(y = K+1|\tilde{x}^1, x^2)\right] \\ + \; & \frac{1}{2}\mathbb{E}_{(x^1,\tilde{x}^2)\sim p_{G_2}}\left[\log p_D(y = K+1|x^1, \tilde{x}^2)\right]. \end{aligned} \tag{5.1}$$

In this way, we stated minmax game over $K+1$ component of the discriminator. In addition to this objective, we made generators also learn from the labels of completed samples. Therefore, the following equation defines objective for the generators $V_{G_{1,2}}(D, G_1, G_2)$:

$$\begin{aligned} \max_{G_1,G_2} V_{G_{1,2}}(D, G_1, G_2) = \; & \frac{1}{2}\mathbb{E}_{(x^2,y)\sim\mathcal{S}_2,z}\left[\log p_D(y|G_1(x^2, z), x^2)\right] \\ + \; & \frac{1}{2}\mathbb{E}_{(x^1,y)\sim\mathcal{S}_1,z}\left[\log p_D(y|x^1, G_2(x^1, z))\right]. \end{aligned} \tag{5.2}$$

Note that, following Assumption 1, we impose the generators to produce equally informative views by assigning the same weight to their corresponding terms in the objective functions (Eq. 5.1, 5.2).

5.4.4 Theoretical Background and Convergence

From the outputs of the discriminator for all $\mathbf{x} \in \mathcal{X}$ we build an auxiliary function $\mathbf{D}(\mathbf{x}) = \sum_{k=1}^{K} p_D(y = k \mid \mathbf{x})$ equal to the sum of the first K outputs associated to the true classes. In this following, we provide a theoretical analysis of Cond^{2}GAN involving the auxiliary function $\mathbf{D}$ under non-parametric hypotheses.

Proposition 1. *For fixed generators G_1 and G_2, the objective defined in (Eq. 5.1) leads to the following optimal discriminator $\mathbf{D}^*_{G_1,G_2}$:*

$$\mathbf{D}^*_{G_1,G_2}(x^1,x^2) = \frac{p_{real}(x^1,x^2)}{p_{real}(x^1,x^2) + p_{G_{1,2}}(x^1,x^2)}, \tag{5.3}$$

where $p_{G_{1,2}}(x^1,x^2) = \frac{1}{2}(p_{G_1}(x^1,x^2) + p_{G_2}(x^1,x^2))$.

Proof. The proof follows from [32]. Let

$$\forall \mathbf{x} = (x^1,x^2), \mathbf{D}(\mathbf{x}) = \sum_{k=1}^{K} D_k(\mathbf{x})$$

From Assumption 2, and the fact that for any observation $\mathbf{x}$ the outputs of the discriminator sum to one i.e. $\sum_{k=1}^{K+1} D_k(\mathbf{x}) = 1$, the value function V_D writes :

$$\begin{aligned} V_D(\mathbf{D},G_1,G_2) = &\iint \log(\mathbf{D}(x^1,x^2)) p_{real}(x^1,x^2) dx^1 dx^2 \\ &+ \frac{1}{2}\iint \log(1-\mathbf{D}(x^1,x^2)) p_{G_1}(x^1,x^2) dx^1 dx^2 \\ &+ \frac{1}{2}\iint \log(1-\mathbf{D}(x^1,x^2)) p_{G_2}(x^1,x^2) dx^1 dx^2 \end{aligned}$$

The equation above can be simplified to the following function form:

$$f(z) = \alpha \log z + \frac{\beta}{2}\log(1-z) + \frac{\gamma}{2}\log(1-z),$$

where $(\alpha,\beta,\gamma) \in \mathbb{R}^3 \backslash \{0,0,0\}$. To find its maximum, we set its derivative to zero, hence:

$$f'(z) = 0 \Rightarrow \frac{\alpha}{z} + \frac{\beta}{2(1-z)} + \frac{\gamma}{2(1-z)} = 0 \Rightarrow z = \frac{\alpha}{\alpha + \frac{1}{2}(\beta+\gamma)},$$

which ends the proof as the discriminator does not need to be defined outside the supports of p_{data}, p_{G_1} and p_{G_2}. Since f has a unique maximizer on the interval of interest, optimal $\mathbf{D}^*_{G_1,G_2}$ is unique as well. □

By plugging back $\mathbf{D}^*_{G_1,G_2}$ (Eq. 5.3) into the value function V_D we have the following necessary and sufficient condition for attaining the global minimum of this function :

Theorem 1. *The global minimum of the function $V_D(G_1,G_2)$ is attained if and only if*

$$p_{real}(x^1,x^2) = \frac{1}{2}(p_{G_1}(x^1,x^2) + p_{G_2}(x^1,x^2)). \tag{5.4}$$

At this point, the minimum is equal to $-\log 4$.

Proof. By plugging back the expression of $\mathbf{D}^*$ (Eq. 5.3), into the value function V_D, it comes

$$V(\mathbf{D}^*, G_1, G_2) = \iint \log\left(\frac{p_{real}(x^1, x^2)}{p_{real}(x^1, x^2) + p_{G_{1,2}}(x^1, x^2)}\right) p_{real}(x^1, x^2) dx^1 dx^2 + \iint \log\left(\frac{p_{G_{1,2}}(x^1, x^2)}{p_{real}(x^1, x^2) + p_{G_{1,2}}(x^1, x^2)}\right) p_{G_{1,2}}(x^1, x^2) dx^1 dx^2$$

Which from the definition of the Kullback Leibler (KL) and the Jensen Shannon divergence (JSD) can be rewritten as

$$\begin{aligned} V_D(\mathbf{D}^*, G_1, G_2) &= -\log 4 + KL\left(p_{real} \,\middle\|\, \frac{p_{real} + p_{G_{1,2}}}{2}\right) + KL\left(p_{G_{1,2}} \,\middle\|\, \frac{p_{real} + p_{G_{1,2}}}{2}\right) \\ &= -\log 4 + 2JSD\big(p_{real} \,\|\, p_{G_{1,2}}\big) \end{aligned}$$

The JSD is always positive and $JSD\big(p_{real} \,\|\, p_{G_{1,2}}\big) = 0$ if and only if $p_{real} = p_{G_{1,2}}$ which ends the proof □

From Equation 5.4, it is straightforward to verify that $p_{real}(x^1, x^2) = p_{G_1}(x^1, x^2) = p_{G_2}(x^1, x^2)$ is a global Nash equilibrium but it may not be unique. In order to ensure the uniqueness, we add the Jensen-Shannon divergence between the distribution p_{G_1} and p_{real} and p_{G_2} and p_{real} the value function V_D (Eq. 5.1) as stated in the corollary below.

Corollary 1. *The unique global Nash equilibrium of the augmented value function :*

$$\bar{V_D}(\mathbf{D}, G_1, G_2) = V(\mathbf{D}, G_1, G_2) + JSD(p_{G_1}||p_{real}) + JSD(p_{G_2}||p_{real}), \tag{5.5}$$

is reached if and only if

$$p_{real}(x^1, x^2) = p_{G_1}(x^1, x^2) = p_{G_2}(x^1, x^2), \tag{5.6}$$

where $V_D(\mathbf{D}, G_1, G_2)$ is the value function defined in Equation (5.1) *and $JSD(p_{G_1}||p_{real})$ is the Jensen-Shannon divergence between the distribution p_{G_1} and p_{real}.*

Proof. The proof follows from the positiveness of JSD and the necessary and sufficient condition for it to be equal to 0. Hence, $\bar{V_D}(\mathbf{D}, G_1, G_2)$ reaches it minimum $-\log 4$, iff $p_{G_1} = p_{real} = p_{G_2}$. □

This result suggests that at equilibrium, both generators produce views such that observations with their completed view follow the same real distribution than those which have their both views observed.

5.5 Experimental Results

In this Section, we present experimental results aimed at evaluating how the generation of views by Cond^{2}GAN can help to take advantage of existing training examples, with many having an incomplete view, in order to learn an efficient classification function.

5.5.1 Experimental Setup

Particularly in our experiments, we focus on the case when the number of training documents having their two versions is much smaller than those with only one of their available versions (i.e. $m_F \ll m_1 + m_2$). This corresponds to the case where the effort of gathering documents in different languages is much less than translating them from one language to another. Accordingly, we randomly select $m_F = 300$ samples having their both views, $m_1 = m_2 = 6000$ samples with one of their views missing and the remaining samples without their translations for testing. The choice of values for m_F, m_1, and m_2 is explained by the size of the collection (and distribution of its classes as well) that we work with described in the following Section 5.5.1.1 and Table 5.1.

5.5.1.1 Data

Table 5.1: The statistics of RCV1/RCV2 Reuters data collection used in our experiments.

Language	# docs	(%)	vocab dim
EN	18,758	16.78	21,531
FR	26,648	23.45	24,893
GR	29,953	26.80	34,279
IT	24,039	21.51	15,506
SP	12,342	11.46	11,547
Total	111,740		

Class	Size (all lang.)	(%)
C15	18,816	16.84
CCAT	21,426	19.17
E21	13,701	12.26
ECAT	19,198	17.18
GCAT	19,178	17.16
M11	19,421	17.39

We perform experiments on a publicly available collection, extracted from Reuters RCV1/RCV2, that is proposed for multilingual multiclass text categorization [1]. The data set contains numerical feature vectors of documents originally presented in five different languages: English (EN), French (FR), German (GR), Italian (IT), and Spanish (SP) (Table 5.1). Documents in different languages belong to one and only one class within the same set of classes ($K = 6$); and they also have translations into all the other languages. These translations are obtained from a state-of-the-art Statistical MT system [70] trained over the Europal parallel collection using about 8.10^6 sentences for the 4 considered pairs of languages.[2]

5.5.1.2 Model and Algorithm Implementation

Architectures of employed neural networks are summarised in Table 5.2. We initialized the generative components of the Cond^{2}GAN as two layers neural networks with one dense hidden layer with a sigmoid activation function and the final dense output layer without any activation. Since the values of the generated samples are supposed to approximate any possible real value, we do not use the activation function in the outputs of both generators.

[1] https://archive.ics.uci.edu/ml/datasets/Reuters+RCV1+RCV2+Multilingual,+Multiview+Text+Categorization+Test+collection

[2] http://www.statmt.org/europarl/

Table 5.2: Description of different specifications used for the neural networks to define each of the components in the Cond^{2}GAN. Here $\texttt{lang\_dim}_v$ denotes a dimension of feature vectors in language that corresponds to the view v. Also note that a dimension for the input in generators is $2\times\texttt{lang\_dim}_v$, since we use noise vector z with the same dimension as the view v respectively.

Component of Cond^{2}GAN	Layer (type)	Input dim	Activation	Output dim
G_1	hidden (dense)	$d_{lang_1} + d_{lang_2}$	sigmoid	200
	output (dense)	200	-	d_{lang_1}
G_2	hidden (dense)	$d_{lang_1} + d_{lang_2}$	sigmoid	200
	output (dense)	200	-	d_{lang_2}
D	hidden (dense)	$d_{lang_1} + d_{lang_2}$	sigmoid	200
	output (dense)	200	sigmoid	7

The discriminator in the Cond^{2}GAN is initialized in the same fashion, expert it includes a sigmoid activation function at the output dense layer as well.

During training, in order to avoid the collapse of the generators [59], we perform minibatch discrimination by allowing the discriminator to have access to multiple samples in combination. From this perspective, the minmax game (Eq. 5.1, 5.2) is equivalent to the maximization of a cross-entropy loss, and we use minibatch training to learn the parameters of the three models. The corresponding empirical errors estimated over a minibatch $\mathcal{B}$ that contains m_b samples from each of the sets $\mathcal{S}_F$, $\mathcal{S}_1$ and $\mathcal{S}_2$ are :

$$\begin{aligned}\mathcal{L}_D(\mathcal{B}) = &-\frac{1}{m_b}\sum_{\mathbf{x}\in\mathcal{B}\cap\mathcal{S}_F}\frac{1}{K+1}\sum_{k=1}^{K} y_k \log\left[D_k(x^1, x^2)\right] \\ &-\frac{1}{2m_b}\sum_{\mathbf{x}\in\mathcal{B}\cap\mathcal{S}_1}\log\left[D_{K+1}(G_1(z^1, x^2), x^2))\right] \\ &-\frac{1}{2m_b}\sum_{\mathbf{x}\in\mathcal{B}\cap\mathcal{S}_2}\log\left[D_{K+1}(x^1, G_2(x^1, z^2))\right]\end{aligned} \tag{5.7}$$

$$\mathcal{L}_{G_1}(\mathcal{B}) = -\frac{1}{m_b}\sum_{\mathbf{x}\in\mathcal{B}\cap\mathcal{S}_2}\frac{1}{K+1}\sum_{k=1}^{K} y_k \log\left[D_k(G_1(z^1, x^2), x^2)\right] + \mathcal{L}^1_{FM} \tag{5.8}$$

$$\mathcal{L}_{G_2}(\mathcal{B}) = -\frac{1}{m_b}\sum_{\mathbf{x}\in\mathcal{B}\cap\mathcal{S}_1}\frac{1}{K+1}\sum_{k=1}^{K} y_k \log\left[D_k(G_2(z^2, x^1), x^1)\right] + \mathcal{L}^2_{FM} \tag{5.9}$$

In order to be inline with the premises of Corollary 1; we empirically tested different solutions and the most effective one that we found was the feature matching technique proposed in [59], which addressed the problem of instability for the learning of generators

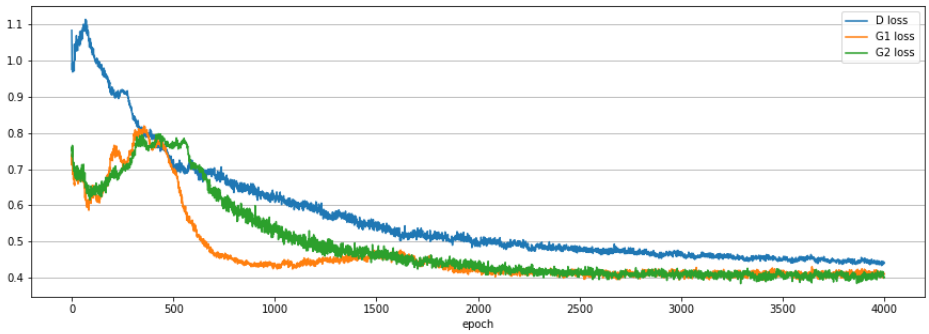


Figure 5.3: Loss of all Cond^{2}GAN components during training for (EN, IT) .

by adding a penalty terms:

$$\begin{aligned} \mathcal{L}^1_{FM} &= \|\mathbb{E}_{p_{real}} f(x^1, x^2) - \mathbb{E}_{p_{G_1}} f(G_v(x^1), x^2)\|, \\ \mathcal{L}^2_{FM} &= \|\mathbb{E}_{p_{real}} f(x^1, x^2) - \mathbb{E}_{p_{G_2}} f(x^1, G_v(x^2))\|, \end{aligned} \tag{5.10}$$

to their corresponding objectives (Eq. 5.8, 5.9). Where, $\|.\|$ is the ℓ_2 norm and f is the sigmoid activation function on an intermediate layer of the discriminator. The stability of the training with described above losses was confirmed experimentally as well (Figure 5.3).

The overall algorithm of Cond^{2}GAN is shown above. The parameters of the three neural networks are first initialized using *Xavier*. For a given number of iterations T, minibatches of size $3m_b$ are randomly sampled from the sets $\mathcal{S}_F$, $\mathcal{S}_1$ and $\mathcal{S}_2$. Minibatches of noise vectors are randomly drawn from the uniform distribution. Models parameters of the discriminator and both generators are then sequentially updated using *Adam* optimization algorithm [39] with its parameters to be set to $\alpha = 10^{-4}$, $\beta = 0.5$. Those settings promise a good convergence of losses (Figure 5.3).

Algorithm 1: Minibatch stochastic training of Cond^{2}GAN

Data: A training set $\mathcal{S} = \mathcal{S}_F \sqcup \mathcal{S}_1 \sqcup \mathcal{S}_2$
Initialize size of minibatches - m_b;
Use $Xavier$ initializer to define initial parameters of discriminator - $\theta_d^{(0)}$;
Use $Xavier$ initializer to define initial parameters of generators - $\theta_{g_1}^{(0)}, \theta_{g_2}^{(0)}$;
for $i = 0 \dots T-1$ **do**
  Sample randomly a minibatch $\mathcal{B}_i$ of size $3m_b$ from $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_F$;
  Create minibatches of noise vector z^1, z^2 from $\mathcal{U}(-1, 1)$;
  $\theta_d^{(i+1)} \leftarrow Adam(\mathcal{L}_D(\mathcal{B}_i), \theta_d^{(i)}, \alpha, \beta)$; /* Update D */
  $\theta_{g_1}^{(i+1)} \leftarrow Adam(\mathcal{L}_{G_1}(\mathcal{B}_i), \theta_{g_1}^{(i)}, \alpha, \beta)$; /* Update G_1 */
  $\theta_{g_2}^{(i+1)} \leftarrow Adam(\mathcal{L}_{G_2}(\mathcal{B}_i), \theta_{g_2}^{(i)}, \alpha, \beta)$; /* Update G_2 */
end

5.5.2 Summary of Results

In our experiments, we consider four pairs of languages with always English as the fist view and one of the rest of the languages as the second; accordingly, four different Cond^{2}GANs are trained with each of data sets $\mathcal{S}_{(\text{EN,FR})}$, $\mathcal{S}_{(\text{EN,SP})}$, $\mathcal{S}_{(\text{EN,IT})}$, and $\mathcal{S}_{(\text{EN,GR})}$. Then for every new observation $\mathbf{x} \in \mathcal{X}_{(\text{EN},l)}$, where $l \in \{\text{FR}, \text{SP}, \text{IT}, \text{GR}\}$, the corresponding Cond^{2}GAN$_{(\text{EN},l)}$ does both: provides a "synthetic" view in case a representation of the sample in one of the languages is missing (by one of the generators), and predicts the class label (by discriminator).

For Cond^{2}GANs performance evaluation, we employ along with them the following classification approaches: one single-view approach, and four multi-view approaches. In the case of using the single-view approach, classifiers are the same as the discriminator and they are trained on the part of the training set with examples having their corresponding view observed. The multi-view approaches are MKL [7], co-classification (co-classif) [4], unanimous vote (mv$_b$) [3]..

In order to evaluate the quality of "synthetic" documents produced by generative components in every model, we focus on comparing the classification scores obtained by other than Cond^{2}GAN multi/single-view classification approaches trained on the very the same pairs of languages. Accordingly, we have two test scenarios:

1. one (denoted by $\mathbf{T}_{\text{EN}l}$) aims to evaluate English documents generation functions in models by pairs of views with English and any other available language ($l \in \{\text{FR}, \text{GR}, \text{IT}, \text{SP}\}$);

2. second (denoted by $\mathbf{T}_{\tilde{\text{EN}}l}$) similarly aims to test on documents generated in another language than English by considering their corresponding English equivalent is provided.

Results are evaluated over the test set using the classification accuracy and the F_1 measure which is the harmonic average of precision and recall. The reported performance are averaged over 20 random train(80%)/test(20%) data splits

5.5.2.1 On the value of the generated views

We start our evaluation by comparing the F_1 scores over the test set, obtained with Cond^{2}GAN and a neural network having the same architecture as the discriminator D of Cond^{2}GAN trained over the concatenated views of documents in the training set where the missing views are generated by MT. Figure 5.4 shows these results.

Each point represents a class, where its abscissa (resp. ordinate) represents the test F_1 score of the Neural Network trained using MT (resp. one of the generators of Cond^{2}GAN) to complete the missing views. All of the classes, in the different language pair scenarios, are above the line of equality, suggesting that the generated views by Cond^{2}GAN provide higher value information than translations provided by MT for learning the Neural Network. This is an impressive finding, as the resources necessary for the training of MT is large (8.10^6 pairs of sentences and their translations); while Cond^{2}GAN does both view completion and discrimination using only the available training data. This is mainly because both

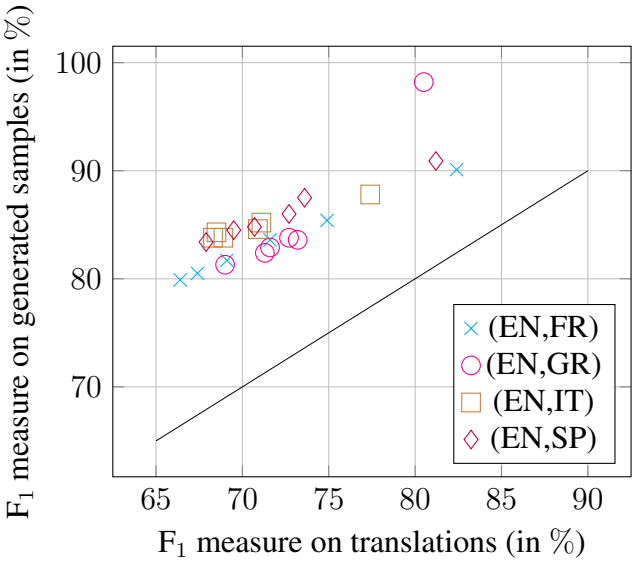


Figure 5.4: F_1-score per class measured for test predictions made by a neural network, with the same architecture than the discriminator of Cond^{2}GAN, and trained over documents where their missing views are generated by MT, or by G_1 or G_2.

generators induce missing views with the same distribution than real pairs of views as stated in Corollary 1.

5.5.2.2 Comparison between multi-view approaches.

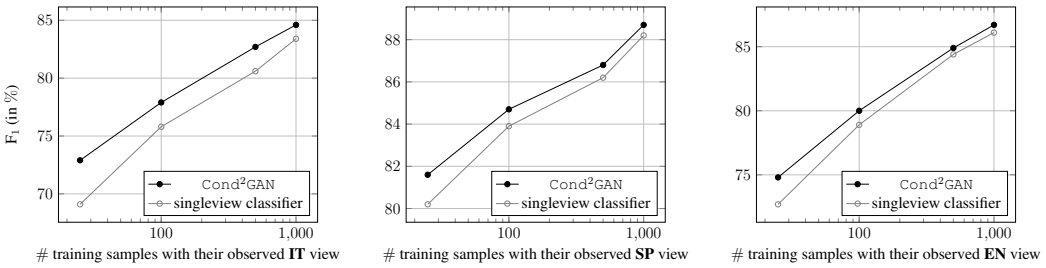
We now examine the gains, in terms of accuracy, of learning the different multiview approaches on a collection where for other approaches than Cond^{2}GAN the missing views are completed by one of the generators of our model. Table 5.3 summarizes these results obtained by Cond^{2}GAN, MKL, co-classif, and mv$_b$ for both test scenarios. In all cases Cond^{2}GAN, provides significantly better results than other approaches. This provides empirical evidence of the effectiveness of the joint view generation and class prediction of Cond^{2}GAN. Furthermore, MKL, co-classif and Cond^{2}GAN are binary classification models and tackle the multiclass classification case with one vs all strategy making them to suffer from class imbalance problem. Results obtained using the F_1 measure are in line with those of Table 5.3 and they are not reported for the sake of space.

5.5.2.3 Impact of the increasing number of observed views.

In Figure 5.5, we compare F_1 measures between Cond^{2}GAN and one of the single-view classifiers with an increasing number of training samples, having the view corresponding to the single-view classifier observed; while the number of training examples with the other observed view is fixed. With an increasing number of training samples, the corresponding single-view classifier gains in performance. On the other hand, Cond^{2}GAN can leverage the lack of information from training examples by having their other view observed, making the difference in performance between these models for a small number of training samples higher.

Table 5.3: Test classification accuracy averaged over 20 random training/test sets. For each of the pairs of languages, the best result is in bold, and a $^{\downarrow}$ indicates a result that is statistically significantly worse than the best, according to a Wilcoxon rank sum test with $p < .01$.

Approaches	(EN,l = FR)		(EN,l = GR)		(EN,l = IT)		(EN,l = SP)	
	$\mathbf{T}_{\text{EN}\tilde{l}}$	$\mathbf{T}_{\tilde{\text{EN}}l}$	$\mathbf{T}_{\text{EN}\tilde{l}}$	$\mathbf{T}_{\tilde{\text{EN}}l}$	$\mathbf{T}_{\text{EN}\tilde{l}}$	$\mathbf{T}_{\tilde{\text{EN}}l}$	$\mathbf{T}_{\text{EN}\tilde{l}}$	$\mathbf{T}_{\tilde{\text{EN}}l}$
MKL	75.6$^{\downarrow}$	77.3$^{\downarrow}$	79.4$^{\downarrow}$	79.6$^{\downarrow}$	78.4$^{\downarrow}$	79.8$^{\downarrow}$	81.2$^{\downarrow}$	83.5$^{\downarrow}$
co-classif	81.4$^{\downarrow}$	83.2$^{\downarrow}$	84.3$^{\downarrow}$	81.6$^{\downarrow}$	82.7$^{\downarrow}$	82.5$^{\downarrow}$	85.1$^{\downarrow}$	86.2$^{\downarrow}$
mv$_b$	83.1$^{\downarrow}$	84.5$^{\downarrow}$	85.2$^{\downarrow}$	79.9$^{\downarrow}$	84.3$^{\downarrow}$	82.1$^{\downarrow}$	84.4$^{\downarrow}$	86.2$^{\downarrow}$
Cond^{2}GAN	**85.3**	**85.1**	**86.6**	**82.9**	**85.3**	**84.5**	**86.5**	**88.3**



(a) $m_{\text{EN}} = 6000$ (b) $m_{\text{EN}} = 6000$ (c) $m_{\text{GR}} = 6000$

Figure 5.5: F_1 measure of Cond^{2}GAN and a single view classifier (c_l) for an increasing number of training samples with the corresponding view that is observed. The number of training examples corresponding to the other view ($m_{\tilde{l}} = 6000$); and the number of training examples with their both views observed is $m_F = 300$.

5.5.2.4 Quality of the generated views.

Moreover, we present some documents in English as well as the top 20 words in the feature characteristics of the generated vector in different languages for each of the documents (Table 5.4). Words that are in the English vocabulary which served for the initial bag-of-word representation of the documents. In this way, it was exhibited qualitatively that generated documents have meaningful translated words bearing similar ideas compared to the original ones; and that, without employing any large external parallel corpora to learn the translations as it would be the case if MT were used.

5.5.2.5 Experiments with MNIST data set

Additionally, the performance of proposed Cond^{2}GAN was tested with MNIST data set. We considered that images can be presented with two halfs as two views, and some images can be corrupted meaning that one of the parts is missing. As a result of training Cond^{2}GAN we got a discriminator that can classify full images with the average accuracy 98.68% which is compatible with the state-of-art methods; generators that can impute missing parts of images (Figure 5.6(b), 5.7(b)), so that discriminator classify them with the average accuracy 94.39% and 94.71% in case left or right part is missing respectively. It gives a prove that

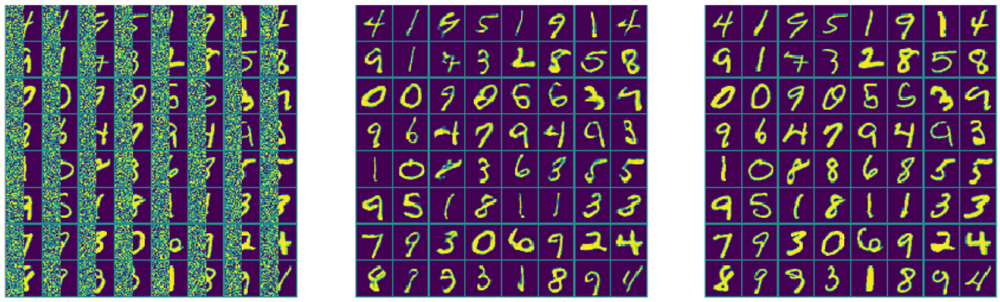
Original documents in English	Top 20 words in the feature characteristics of generated vectors by Cond^{2}GAN
Fleet Financial Group and National Westminster Bank PLC said Tuesday they signed an agreement that will allow both companies to provide banking services to corporate customers in Britain and the United States. Under the agreement, NatWest will set up a representative office in Boston to provide Sterling and foreign currency account and cash management services to American companies that either have a physical presence in Britain or trade there. Desks also will be created in Boston, New York and London, where each bank's customers can receive quick help in opening accounts and cash management. The offices will be staffed by employees who will bring specific expertise in their country's banking system to each marketplace, the companies said. Fleet will provide U.S. dollar accounts and cash management services to the U.S. subsidiaries and offices of British-based companies. Fleet currently markets U.S. cash management services directly to British companies from its office in London. NatWest supplies commercial banking services to about one-third of the companies in Britain and is the second largest retail bank in the country with over 2,000 branches.	FR: succursale gestion société bureau marché trésorerie échange commerciale accord service bancaire client banque filiale permettre entreprise système signature devise compte GR: konto zweig management tochtergesellschaft unternehmen büro markt fiskus austausch kommerziell währung vereinbarungservice bankwesen kunde bank erlauben unternehmen system unterschrift SP: sucursal oficina efectivo acuerdo intercambio comercial administración servicio bancario cliente banco filial compañía estadounidense empresa firma americano británico moneda cuenta IT: accordo servizio bancario cliente banca ufficio commerciale filiale mercato contante scambio società controllata azienda americana gestione conto valuta personale clienti
World oil prices eased on Tuesday in a market where refineries stung by high crude oil premiums and poor margins began to buy a cheaper barrel. The North Sea World Benchmark October Brent Blend crude oil futures closed 38 cents at 20.43 a barrel, after not exceeding the daily high of 20.80. There was a general feeling in the marketplace that Brent was overheated and a trader had to say. On the unofficial futures market for Brent, the timing or physical differences for Brent declined, suggesting that cargoes would earn lower premiums in the coming weeks. This could avert the risk that refineries use less crude oil through their systems to increase the price of their products. The market was also waiting for instructions from US stocks to be released later on Tuesday. Fuel oil and diesel reserves are predicted to increase by 1.3 million barrels in the run up to the winter season, as gasoline fuel oil will soften the lead for crude oil prices. Low inventories in the United States largely supported the markets in the North Sea and West Africa. In the last week alone, 5.0 million barrels of distilled North Sea varieties were on their way across the Atlantic.	FR: pétrole prix marché brut baril commercant diminué cargaison diesel réserve prédit augmenter hiver saison essence carburant distillé mer produit GR: diesel reservieren ol preis markt raffinerien roh fass handler abgelehnt ladung vorhergesagt erhohen ansteigen jahreszeit benzin treibstoff destilliert meer produkt SP: comerciante invierno temporada gasolina rechazado carga diesel reserva petróleo precio mercado refinerias crudo barril predicho incrementar combustible destilado instrucción producto IT: carburante distillata istruzione prodotto raffinerie greggio barile commerciante diminuito carico diesel riserva olio prezzo mercato previsto aumentare inverno stagione gasolio

Table 5.4: Example of the generated views.

Cond^{2}GAN can be successfully applied with data sets of images.

(a) (b) (c)

Figure 5.6: Examples of pictures with right half missing (a) that were imputed by G_1 (b) and result can be compared with real images (c).



(a) (b) (c)

Figure 5.7: Examples of pictures with left half missing (a) that were imputed by G_2 (b) and result can be compared with real images (c).

5.5.2.6 Results in Virtual Metrology

As anticipated in Section 4.3.4 and specifically in Section 4.3.4.1, one of the biggest drivers toward better generalization is the ability to provide the design characterization. However, such measurements are too expensive to extract. Therefore, for the use case study described in Section 4.3.4.1 we were given the feature characteristics for only 10 of the most populated designs out of 146 total products given in the full data collection. Then it was proven that the presence of the predictor variables describing the design allowed boosting the predictability as high as 30% of increase in the R^2 for unseen during training products.

In this Section, we consider the rest of the products with their design features missing. For them, we investigate the possibility of completing their design characteristics by the Cond^{2}GAN. In particular, the two views are the product features, and the ESD extracted features where the first view may be completely missing during training or testing.

As Cond^{2}GANis proposed for the joint learning of both missing view imputation and target prediction, that is the classification task, while the VM task is initially defined as a regression one, we used a target discretization to define categories. Then, instead of

predicting a numeric value, the model estimates the probability that a sample belongs to a set of fixed bins, where the acceptable error rate defines the size of the bin. Particularly, the CuECD thickness is distributed between 8 and 12, and the acceptable error rate defined by the enterprise is 0.2; accordingly, 20 classes were created, which finishes the preparation setup for the Cond^{2}GAN.

The quality of the completed data by the Cond^{2}GANis verified by the comparison of the predictability on new products in the Test set of the GBDT-based VM framework proposed in the Section 4.3. In the first experiment, the model is trained with the data set where most of the observations have their product information missing. Next, we use a set where most of the observations have missing product information. Next, we use Cond^{2}GANto complete the missing information in the training and test data and train the VM framework to compare the results with the previous performance. This result is reported in Table 5.5, which shows that Cond^{2}GAN-bases imputation allows boosting the predictability as high as 13% of increase in the R^2 for unseen during training products.

Table 5.5: Predictability of GBDT model on the Test set of the CuECD VM use case for the observations that have their product information missing.

Approaches	Test set where product features missing	
	R^2	RMSE
GBDT (with no imputation)	-0.0231	0.1536
Cond^{2}GAN+ GBDT	**0.1146**	**0.1459**

5.6 Discussion

In this Chapter, we presented Cond^{2}GAN for multi-view multi-class classification where observations may have missing views. The model consists of three ANNs implementing a three players game between a discriminator and two generators. For an observation with a missing view, the corresponding generator produces the view conditionally on the other observed one. The discriminator is trained to recognize observations with a generated view from others having their views complete and to classify the latter into one of the existing classes. Experiments on a subset of Reuters RCV1/RCV2 show the effectiveness of Cond^{2}GAN to generate high-quality views allowing to achieve significantly better results, compared to the case where the missing views are generated by Machine Translation which requires a large collection of sentences and their translations to be tuned.

Chapter 6

A missing data imputation approach based on conditional GANs applied to a real challenging EHR dataset

Similarly to the SM industry, the missing data is a relevant and established problem also in biomedical informatics communities. Several real-world Electronic Health Record (EHR) data sets comprise several missing values, thus revealing a high level of spatio-temporal sparsity in the predictors' matrix. In light of these factors, the following submitted paper presents a data imputation method based on a clinical conditional Generative Adversarial Network (ccGAN) that can impute missing values by using non-linear and multivariate data from various patients [BDaEFA22].

6.1 Motivation

Given the increasing and unavoidable digital transformation process of national healthcare system management, the huge size of structured EHR data is beginning to be available. In predictive and precision medicine, Machine Learning (ML) techniques are capable of managing real EHR data and providing disease predictions. On the other hand, the potential of ML may be limited from the low quality of the EHR data, i.e. high sparsity, imbalanced setting, noisy and redundant features, and irregular time sampling characteristics. This challenging scenario is emphasized in routine EHR data (i.e., general practitioners, diabetic centers, clinics) where not all laboratory exams are prescribed uniformly over time. Given these reasons, an adequate and effective missing data imputation stage assumes crucial importance within the data preprocessing pipeline. Specifically, a suitable data imputation strategy may positively influence the effectiveness of the ML algorithm for prognosis and disease prediction.

This study seeks to offer a data imputation technique based on a clinical conditional Generative Adversarial Network (ccGAN) capable of imputing missing values of observed characteristics conditioned by fully-available characteristics values to be then employed for predicting the probable diabetes complication.

We investigate our proposed strategy via the lens of a specific clinical use case (i.e.,

diabetic retinopathy (DR) prediction) of diabetes complications. DR caused by chronically high or variable blood sugar is the most typical and insidious diabetes microvascular complication. With the worldwide increasing incidence of diabetic patients with DR and consequential visual impairments, early diagnosis of DR and timely appropriate treatment are progressively becoming an effective measure to prevent DR and alleviate the economic burden over the national healthcare systems [58]. Physicians typically diagnose the DR through by directly evaluating fundus images, but this gold standard process, usually carried out when the DR has already been delineated, remains expensive, time-consuming, and sometimes unnecessary [55]. Thus, the early prediction of developing DR by employing only routine EHR data and ML techniques may result in a convenient and effective strategy for follow-up diabetic patients within a screening scenario.

6.2 Contribution

The main contributions to biomedical informatics are threefold and can be summarized as follows:

- we propose a ML approach to impute missing values from EHR data and provide the prediction of DR. The data imputation strategy is based on a novel ccGAN architecture that exploits the fully-available clinical features among different patients to infer other missing clinical features. The prediction phase is realized by implementing and comparing different ML classifiers;
- we evaluate the quality of the imputed values predicted by ccGAN versus other state-of-the-art GAN-based missing data imputation strategies;
- we show how the proposed ccGAN approach overcomes other state-of-the-art data imputation strategies to solve disease prediction tasks using a real challenging EHR dataset. Moreover, the employed ML models may support the clinician by revealing the most discriminative features by also taking into account the missing values.

6.3 EHR Dataset

The missing data mechanism can be categorized into the following three cases: completely at random, at random, or not a random [76]. In our case, we provide results under the missingness completely at random (MCAR) assumption. Moreover, experimental results are also provided under the real-clinical scenario where laboratory exams are not prescribed uniformly over time.

The EHR data, we worked with during this project, consist of 120K diabetic patients and are structured in *demographics field* (i.e., patient's identificative number (ID patient), gender, year of birth, diabetes diagnosis date); *pathological field* (i.e., ID patient, ICD-9 codes, pathology diagnosis date); *lab tests field* (i.e., ID patient, lab tests codes, lab tests values, lab tests prescription date).

6.3.1 Definition of control and DR patients

The diabetologist selected all the ICD-9 codes *pathological field* associated with DR: the univocal ICD-9 code indicates a non-DR condition, while all the other ICD-9 codes indicate a DR condition.

All the ICD-9 codes that did not specify DR or non-DR conditions were removed from *pathological field*. Then, for every patient, both ICD-9 and lab tests codes were removed if pathology diagnosis date and lab tests prescription date preceded the diabetes diagnosis date. Figure 6.1 describes the inclusion criteria to select the time-window of interest (TWOI) for both control and DR patients.

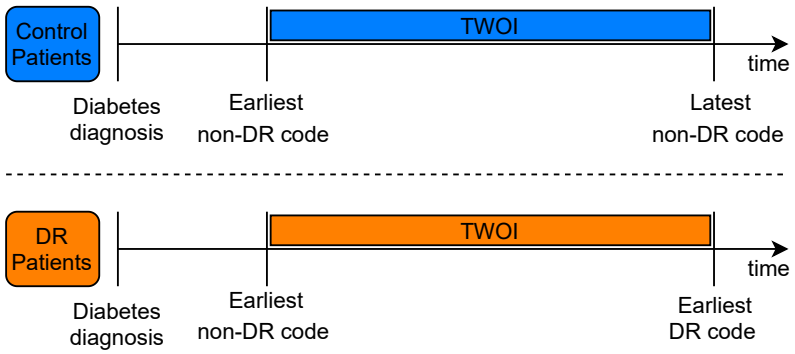


Figure 6.1: Observational time window of interest (TWOI) for control and DR patients.

Control patients - TWOI: A control patient was defined by at least two consecutive ICD-9 codes of non-DR and none of the DR codes within the TWOI. A TWOI of a control patient (see Figure 6.1 - upper side) is delimited by the earliest ICD-9 code of non-DR and the latest ICD-9 code of non-DR.

DR patients - TWOI: A DR patient was defined by at least an ICD-9 code of non-DR followed by one ICD-9 code of DR. A TWOI of a DR patient (see Figure 6.1 - bottom side) is delimited by the earliest ICD-9 code of non-DR and the earliest ICD-9 code of DR. A patient was included in the study only if the date of the earliest ICD-9 code of non-DR preceded the earliest date of ICD-9 code of DR.

6.3.2 Preprocessing

Following the definition of control and DR patients, the EHR data consists of 40555 patients (31611 control patients, 8944 DR patients) and 60 demographical and lab tests features (predictors). The preprocessing procedure consists of features analysis and patient selection stages.

Features analysis

A subset of 48 predictors was chosen by two diabetologists, based on their experience in the clinical task of interest. Thus, the predictors were grouped by the distribution of their missing values (see Figure 6.3). Predictors were split in green (X_g), yellow (X_y) and red (X_r) predictors in according to the following criteria (see Figure 6.2):

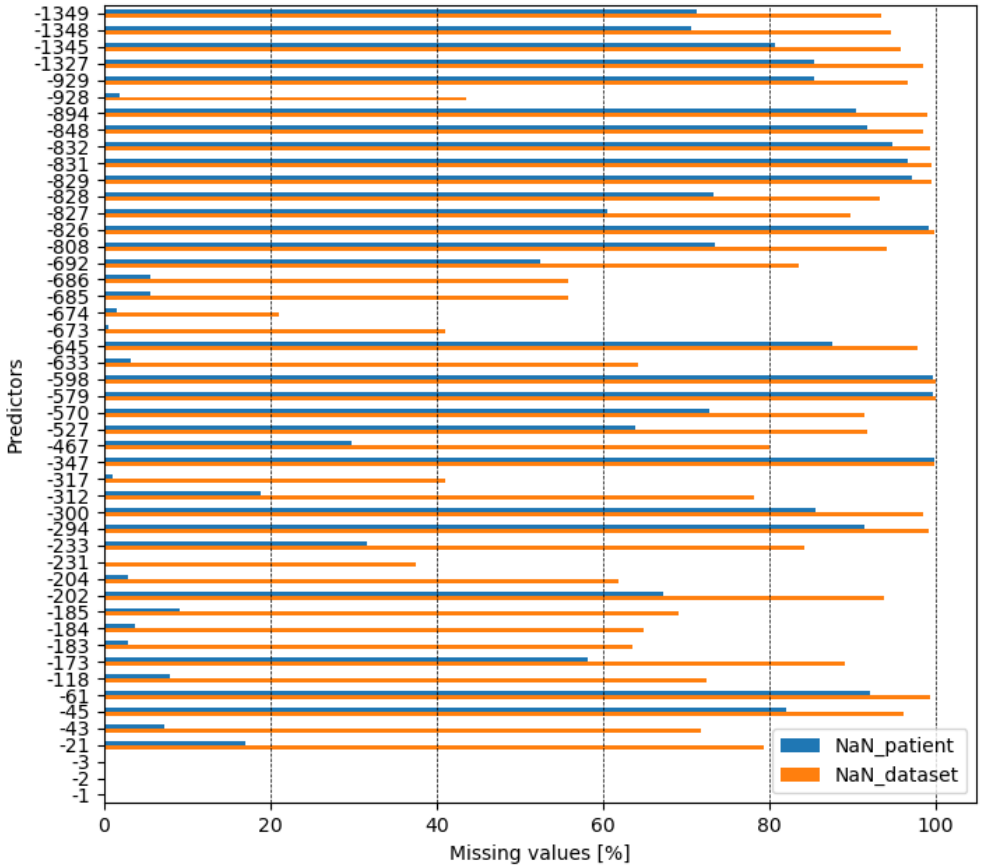


Figure 6.2: Missing values (NaNs) distribution over patients (blue) and over the whole EHR dataset (orange).

- X_g contains less than 2% of missing values per patient and less than 50% of missing values for the whole dataset;
- X_y contains between 3% and 40% of missing values per patient and between 50% and 80% of missing values for the whole dataset;
- X_r contains more than 40% of missing values per patient and more than 80% of missing values for the whole dataset.

Patient selection

In order to obtain the X_g predictors fully filled (i.e., no missing values) across all the patients, we removed the 2981 patients (i.e., $\sim 80\%$ control patients, 20% DR patients) that do not contain simultaneously all the X_g predictors. Table 6.1 describes the statistics of the EHR data after the patient selection preprocessing stage.

-1349	Albumin to creatinine ratio	mg/mmol
-1348	Creatinine clearance	ml/min
-1345	Creatininuria	mg/dl
-1327	Winsor index	Null
-929	Microalbuminuria	mg/24h
-928	Body mass index	Kg/m^2
-894	Urine culture	Null
-848	Potassium (uri)	mEq/l
-832	Pre-prandial glycaemia	mg/dl
-831	Pre-dinner glycaemia	mg/dl
-829	Glycaemia h 23	mg/dl
-828	Post-prandial glycaemia	mg/dl
-827	Post-breakfast glycaemia	mg/dl
-826	Post-dinner glycaemia	mg/dl
-808	Creatinine clearance	ml/min
-692	Urine ketones	mg/dl
-686	Diastolic pressure	mmHg
-685	Systolic pressure	mmHg
-674	Height	cm
-673	Weight	kg
-645	Urea	mg/dl
-633	12-hour fasting triglycerides	mg/dl
-598	Sodium (uri)	mEq/l
-579	Albuminuria/creatinuria ratio	Null
-570	Proteines (uri)	mg/dl
-527	Blood plates	$1000/mm^3$
-467	Microalbuminuria	mg/l
-347	Glicosuria	G/l
-317	Fasting glycaemia	mg/dl
-312	Gamma-glutamyl transferase	UI/l
-300	Alkaline phosphatase	UI/l
-294	Fibrinogen (serum)	mg/dl
-233	Hemoglobin	g/dl
-231	Glycated hemoglobin	%
-204	Creatinine	mg/dl
-202	Creatine phosphokinase (serum)	UI/l
-185	LDL cholesterol	mg/dl
-184	HDL cholesterol	mg/dl
-183	Cholesterol (total)	mg/dl
-173	Weist	cm
-118	Serum glutamic-oxaloacetic transaminase	UI/l
-61	Amylase	UI/l
-45	Albumin excretion rate	mcg/min
-43	Alanine aminotransferase test	UI/l
-21	Uric acid	mg/dl
-3	Gender	Null
-2	Age	years
-1	Diabates duration	years

Figure 6.3: Green predictors (X_g) indicate a very low presence of missing values, yellow predictors (X_y) indicate a mild presence of missing values, and red predictors (X_r) indicate a high presence of missing values according to the criteria defined.

Table 6.1: Statistics of the EHR dataset.

Description	**Statistics**
Total patients	37574
Control:	78%
DR:	22%
Gender	
Male:	56%
Female:	44%
Age ($years$)	$68(\pm 12)$
Diabetes duration ($years$)	$12(\pm 8)$
# of observations per patient	$19(\pm 15)$
Predictors	48
X_g:	8
X_y:	13
X_r:	27

6.4 Method

The amount of observations for each patient in our real clinical EHR dataset is limited and sparsely distributed over time (see Table 6.1). By computing the auto-distance correlation and basing our decision on this supporting data, we attempt to rule out the possibility of temporal connections between different observations of the same patient throughout time (see Section 6.4.1). Afterward, we presented our ccGAN for data imputation on the selected EHR dataset.

6.4.1 Auto-distance correlation function

Auto-distance correlation function (ADCF) measures temporal correlation accross univariate time series [79]. The ADCF can be expressed as a V-statistic of order two, which under the null hypothesis of independence is degenerate. Thus, considering a traditional autocorrelation plot where the confidence intervals are got simultaneously, may turn to be a complex task. Given this motivation, the $(1-\alpha)\%$ confidence intervals are computed simultaneously adopting the Monte Carlo simulation and the independent wild bootstrap approach [31]. We set the significance level $\alpha = 0.05$ and the number of bootstrap replications $b = 499$ to obtain the $(1-\alpha)\%$ empirical critical values. By exploring different lags (MaxLag=$5, 10, 15, 20, 25$) for computing the ACDF function within multiple observations of the same patient, we did not find any feature that overcome the critical value for more than the 5% of patients. Thus, the non-temporal correlation among the values of the predictors was evidenced.

By taking into account this finding, we employed a non-temporal configuration of the EHR dataset, where a single value for each predictor is considered for the i-th patient.

6.4.2 Clinical conditional Generative Adversarial Network (ccGAN)

In the standard cGAN formulation, two players minimax game between generative neural network G (generator) and discriminative neural network D (discriminator) is defined as following:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\mathbf{x},\mathbf{y}\sim p_{data}(\mathbf{x},\mathbf{y})}\big[\log D(\mathbf{x},\mathbf{y})\big] + \mathbb{E}_{\mathbf{z}\sim p_z(\mathbf{z}),\mathbf{y}\sim p_{data}(\mathbf{y})}\big[\log(1-D(G(\mathbf{z}|\mathbf{y}),y))\big]. \tag{6.1}$$

G and D, both conditioned on some extra information $\mathbf{y}$, are trained simultaneously. Generator learns a function that perform mapping for $\mathbf{z}$ from simple distribution like $\mathcal{U}(0,1)$ to the distribution defined by data collection p_{data}. Thus, generator's objective is to learn to produce samples indistinguishable form real data observations. In contrary, discriminator's objective is to accurately separate generated samples from the real data. Let $\mathbf{x}^g$, $\mathbf{x}^y$ and $\mathbf{x}^r$ are samples of green, yellow and red predictors that are taking values in $\mathcal{X}^g = \mathcal{X}^g_1 \times ... \times \mathcal{X}^g_{d_g}$, $\mathcal{X}^y = \mathcal{X}^y_1 \times ... \times \mathcal{X}^y_{d_y}$ and $\mathcal{X}^r = \mathcal{X}^r_1 \times ... \times \mathcal{X}^r_{d_r}$ spaces respectively. Distribution of the random variables $\mathbf{x}^g$, $\mathbf{x}^y$ and $\mathbf{x}^r$ are defined by corresponding data collections X_g, X_y, X_r and will denote $P(X_g)$, $P(X_y)$, $P(X_r)$. Taking into account the highly limited amount of information provided by the matrix X_r (more than 40% of patients do not contain these features), we decided to impute only X_y predictors, conditioned on extra information given by X_g predictors. It is worth noting that, differently from the state-of-the-art literature the imputation of X_y still represents a challenging task, where the available information is highly limited and the proposed ccGAN approach should accurately impute between the 3% and 40% of missing values per patient. Accordingly, we consider a data collection $S = \{(\mathbf{x}^y_i, \mathbf{x}^g_i)\}_{i=1}^N = S_F \cup S_\perp$ of size N that consists of two subsets $S_F = \{(\mathbf{x}^y_i, \mathbf{x}^g_i) \in \mathcal{X}^y \times \mathcal{X}^g\}_{i=1}^{N_F}$ and $S_\perp = \{(\mathbf{x}^y_i, \mathbf{x}^g_i) \in \tilde{\mathcal{X}}^y \times \mathcal{X}^g\}_{i=1}^{N_\perp}$, where $\tilde{\mathcal{X}}^y = (\mathcal{X}^y_1 \cup \{\perp\}) \times ... \times (\mathcal{X}^y_{d_y} \cup \{\perp\})$ and symbol $\perp$ indicates unobserved components ($N = N_F + N_\perp$ and $N_F \ll N_\perp$). Then further in our explanation, when referring to a sample $(\mathbf{x}^y, \mathbf{x}^g)$ drawn from $S_\perp$, we will simply use $(\tilde{\mathbf{x}}^y, \mathbf{x}^g)$.

Architecture for our ccGAN-based imputation strategy is shown in Figure 6.4 and consists of two-players neural networks. First is a $G : \tilde{\mathcal{X}}^y \times \mathcal{X}^g \times \{0,1\}^{d_y} \to \mathcal{X}^y$ - generative neural network - that, conditionally on extra information given by green predictors $\mathbf{x}^g$ and partially available values of yellow predictors $\tilde{\mathbf{x}}^y$, performs mapping for random variable $\mathbf{z}$ from distribution $\mathcal{U}(0,1)$ to corresponding complete vector $\mathbf{x}^y_{gen}$. Accordingly, if $\mathbf{m} \in \{0,1\}^{d_y}$ is a mask vector that indicates an availability of each predictor value in $\tilde{\mathbf{x}}^\mathbf{y}$, then

$$\mathbf{x}^y_{gen} = G(\mathbf{m} \odot \tilde{\mathbf{x}}^y, \overline{\mathbf{m}} \odot \mathbf{z}, \mathbf{x}^g),$$

where $\overline{\mathbf{m}}$ denotes a complement of $\mathbf{m}$. Since output of G consist of predictions of even non-missing values, the imputed vector is

$$\hat{\mathbf{x}}^y = \mathbf{m} \odot \tilde{\mathbf{x}}^y + \overline{\mathbf{m}} \odot \mathbf{x}^y_{gen}.$$

Next, similarly to cGAN, we define a discriminative neural network $D : \mathcal{X}^y \times \mathcal{X}^g \to [0,1]$ - an adversary to train G - which objective is to distinguish real full observations $(\mathbf{x}^y, \mathbf{x}^g) \in S_F$ from incomplete but imputed by G observations $(\hat{\mathbf{x}}^y, \mathbf{x}^g)$, where $(\tilde{\mathbf{x}}^y, \mathbf{x}^g) \in S_\perp$. Particularly,

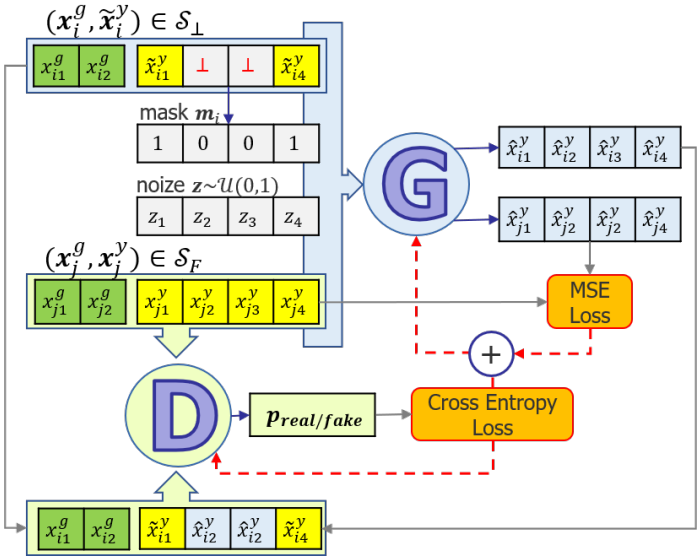


Figure 6.4: The proposed ccGAN architecture.

D and G are trained jointly in a way that D is optimized to maximize a probability of D predicting a correct label for real or synthetic sample, while G is optimized to minimize a probability of D to identify generated samples. Then, discriminative loss for minimax GAN optimization problem in ccGAN model is the following:

$$\begin{aligned} L_d(G,D) = &\mathbb{E}_{(\mathbf{x}^y,\mathbf{x}^g)\in S_F}\big[\log D(\mathbf{x}^y,\mathbf{x}^g)\big] \\ &+ \mathbb{E}_{(\tilde{\mathbf{x}}^y,\mathbf{x}^g)\in S_\perp,\mathbf{z}}\big[\log(1-D(G(\tilde{\mathbf{x}}^y,\mathbf{x}^g,\mathbf{m},\mathbf{z}),\mathbf{x}^g)\big]. \end{aligned} \tag{6.2}$$

Moreover, by taking into account that in our setup $N_F \neq 0$ and data is missing completely at random (MCAR), we use an additional term in the objective function - masked reconstruction loss - computed over real full samples in order to stabilise training of the introduced model. Specifically, lets define an operator f_{nan} that introduce missing values to the full vector of yellow predictors $\mathbf{x}^y$ from $(\mathbf{x}^y,\mathbf{x}^g) \in S_F$ with respect to the mask $\mathbf{m}$:

$$f_{nan}(\mathbf{x}^y,\mathbf{m}) = \mathbf{x}^y \odot \mathbf{m} + nan \odot \overline{\mathbf{m}}.$$

Accordingly, if $\mathbf{m}$ is sampled from the collection of masks in $S_\perp$, which is MCAR, then $(f_{nan}(\mathbf{x}^y,\mathbf{m}),\mathbf{x}^g) \in \tilde{\mathcal{X}}^y \times \mathcal{X}^g$, where $(\mathbf{x}^y,\mathbf{x}^g) \in S_F$; and masked reconstruction loss is defined by:

$$L_r(G) = ||G(f_{nan}(\mathbf{x}^y,\mathbf{m}),\mathbf{x}^g,\mathbf{m},\mathbf{z}) \odot \overline{\mathbf{m}} - \mathbf{x}^y \odot \overline{\mathbf{m}}||_2. \tag{6.3}$$

Finally, in ccGAN imputation strategy two players minimax game between generator G and discriminator D is defined by two-part loss:

$$\min_G \max_D \Big(L_d(G,D) + L_r(G)\Big), \tag{6.4}$$

which we solve in a minibatch stochastic iterative manner described in Algorithm 2. Proposed method shares with the original GAN a property that global minimum is achieved if and only if $p_{data}(\mathbf{x}^y,\mathbf{x}^g) = p_g(\mathbf{x}^y_{gen},\mathbf{x}^g)$, which can be proven as shown in [32].
X_g and the imputed X_y represent the predictors for each patient, while the label is represented in terms of control (0) and DR (1) patients.

Algorithm 2: Pseudo-code of ccGAN.

Data: training set $S = S_F \cup S_\perp$
Initialization: $\theta_D^{(0)}, \theta_G^{(0)}$ # weights for G and D respectively
for $i = 0, \ldots, N_{epochs}$ **do**
  Draw minibatch $\mathcal{B}_F = \{\mathbf{x}_j^y, \mathbf{x}_j^g\}_{j=1}^{m_b}$ from S_F
  Draw minibatch $\mathcal{B}_\perp = \{\tilde{\mathbf{x}}_j^y, \mathbf{x}_j^g\}_{j=1}^{m_b}$ from $S_\perp$
  for $\mathcal{B}_\perp$ **do**
    $\mathbf{m}_j \leftarrow 1 - \mathbb{1}_\perp(\tilde{\mathbf{x}}_j^y)$
    $\hat{\mathbf{x}}_j^y \leftarrow G\big(\mathbf{m}_j \odot \tilde{\mathbf{x}}_j^y + \overline{\mathbf{m}}_j \odot \mathbf{z}_j, \mathbf{x}_j^g\big)$
    $\hat{\mathbf{x}}_j^y = \mathbf{m}_j \odot \tilde{\mathbf{x}}_j^y + \overline{\mathbf{m}}_j \odot \hat{\mathbf{x}}_j^y$
  end
  $L_D = \sum_{\mathcal{B}_F} \log D(\mathbf{x}_j^y, \mathbf{x}_j^g) + \sum_{\mathcal{B}_\perp} \log(1 - D(\hat{\mathbf{x}}_j^y, \mathbf{x}_j^g))$
  $L_G = \sum_{\mathcal{B}_\perp} \log(1 - D(\hat{\mathbf{x}}_j^y, \mathbf{x}_j^g))$
  for $\mathcal{B}_F$ **do**
    $\hat{\mathbf{x}}_j^y \leftarrow G\big(\mathbf{m}_j \odot \mathbf{x}_j^y + \overline{\mathbf{m}}_j \odot \mathbf{z}_j, \mathbf{x}_j^g\big)$
  end
  $L_G = L_G + \frac{1}{m_b} \sum_{\mathcal{B}_F} (\overline{\mathbf{m}}_j \odot \hat{\mathbf{x}}_j^y - \overline{\mathbf{m}}_j \odot \mathbf{x}_j^y)^2$
  $\theta_D^{(i+1)} \leftarrow Adam(-L_D, \theta_D^{(i)}, \alpha, \beta)$ # update of D
  $\theta_G^{(i+1)} \leftarrow Adam(L_G, \theta_G^{(i)}, \alpha, \beta)$ # update of G
end

6.5 Experimental Results

In this section we introduce the comparisons in terms of data imputation and classification ML models techniques.

6.5.1 Experimental Comparisons

Data Imputation Techniques

We start experimental analysis by comparing the quality of imputed values predicted by $ccGAN$ method versus other state-of-the-art GAN-based missing data imputation strategies - baselines like GAIN and MisGAN. In this experiment, all the algorithms are trained with $S_\perp$ set together with randomly selected subset of S_F. The rest of full observations, in their turn, form a set for testing. In other words, $S_F = S_F^{train} \cup S_F^{test}$, where we set train size proportion equal to 0.8. Then, after the model of choice g^* is trained, an accuracy on the test set is evaluated by computing masked mean squared error (MSE) between estimated values of missing yellow predictors in a set $\{(f_{nan}(\mathbf{x}_i^y, \mathbf{m}), \mathbf{x}_i^g)\}_{(\mathbf{x}_i^y, \mathbf{x}_i^g) \in S_F^{test}}$ and their real values:

$$\frac{1}{\|S_F^{test}\|} \sum_{(\mathbf{x}_i^y, \mathbf{x}_i^g) \in S_F^{test}} \big(g^*(f_{nan}(\mathbf{x}^y, \mathbf{m}), \mathbf{x}^g) \odot \overline{\mathbf{m}} - \mathbf{x}^y \odot \overline{\mathbf{m}}\big)^2, \tag{6.5}$$

where m is sampled from the set of masks in $S_\perp$.

Moreover, once evaluated the ccGAN in terms of data imputation performance with respect to the state-of-the-art GAN-based missing data imputation strategies, we compared our proposed ccGAN with other state-of-the-art data imputation techniques such as KNN [49], MissF [62], and MICE [42]. In this case, our goal was to compare performance accuracy of the target label prediction in case of training on complete data imputed by different imputation strategies. This experimental setup is explained next.

ML models, metrics, and experimental procedure

In the prediction stage, we used state-of-the-art ML models widely adopted for disease prediction [10], such as eXtreme Gradient Boosting (XGB), RF, DT, Linear and Gaussian SVM, LR, KNN. The predictive performance was evaluated by the following metrics: Accuracy, macro-F1 (F1), macro-Precision (Precision), macro-recall (Recall), Area Under the receiver operating characteristic Curve (AUC), and Area Under the Precision-Recall Curve (PRAUC). We implemented a 10-fold Cross Validation (CV-10) experimental procedure. The hyperparameters of the ML models were tuned in a nested Fivefold Cross-Validation by implementing a grid-search [15] and optimizing the PRAUC metric.

6.5.2 Imputation Performance

According to the result of averaged masked MSE computed over 20 different random splits of S_F, ccGAN outperforms the baseline models GAIN and MisGAN (see Table 6.2).

Table 6.2: Predictive performance in terms of masked MSE of different GAN-based models for missing data imputation and proposed ccGAN model averaged over 20 random training/test data splits.

Imputation Model	**masked MSE**
GAIN	0.192 ± 0.018
MisGAN (the imputer)	0.203 ± 0.015
ccGAN	$\mathbf{0.154 \pm 0.015}$

The imputation performance comparisons highlighted how the proposed ccGAN strategy was a reliable solution with respect to baseline GAN-based strategies for imputing missing values in the EHR dataset under the MCAR assumption. Next, we present the results of the proposed ccGAN in terms of predictive performance.

6.5.3 Predictive Performance

The XGBoost and RF achieved the best performance among other tested supervised classifiers, such as DT, SVM, LR and KNN. Thus, we show the predictive performance of the proposed data imputation approach by applying XGB (see Table 6.3) and RF (see Table 6.4) as ML classification models. It is worth noting that we exploited the proposed ccGAN for imputing the value of X_y predictors. The overall best predictive performance was reached by the RF model in X_g+X_y setting adopting our proposed ccGAN imputation technique

(PRAUC = 66.16 ± 1.09). The best imputation technique competitor is represented by MICE (PRAUC = 65.53 ± 1.04). The employment of single X_y or X_g predictors leads to a decrease in performance. The same trend was reached by the XGB model but with globally lower predictive performance than RF model. However, ccGAN imputation technique in X_g+X_y setting (PRAUC = 65.20 ± 1.09) keeps on remaining the best strategy.

Table 6.3: Predictive performance of XGB model in X_g, X_y, and $X_g + X_y$ settings: comparison between our proposed data imputation techniques and other competitors. Best predictive performance result in terms of PRAUC is reported in bold.

Predictors	Accuracy	F1	Precision	Recall	AUC	PRAUC
X_y (KNN)	83.12 ± 0.39	66.95 ± 0.86	80.51 ± 1.18	64.06 ± 0.67	76.09 ± 0.68	58.16 ± 1.22
X_y (missF)	82.88 ± 0.35	66.90 ± 0.73	79.14 ± 1.06	64.10 ± 0.57	76.27 ± 0.76	58.44 ± 0.87
X_y (MICE)	82.79 ± 0.38	67.20 ± 0.91	78.25 ± 0.93	64.43 ± 0.74	77.53 ± 0.73	59.41 ± 1.40
X_y **(ccGAN)**	83.06 ± 0.47	67.85 ± 1.07	78.89 ± 1.21	64.96 ± 0.88	77.89 ± 0.55	60.25 ± 1.22
X_g+X_y (KNN)	83.66 ± 0.42	69.18 ± 0.86	80.36 ± 1.21	66.04 ± 0.71	79.69 ± 0.69	62.85 ± 1.06
X_g+X_y (missF)	83.78 ± 0.41	69.59 ± 0.83	80.41 ± 1.08	66.42 ± 0.68	80.08 ± 0.70	63.35 ± 1.23
X_g+X_y (MICE)	83.81 ± 0.55	69.89 ± 0.98	80.16 ± 1.54	66.73 ± 0.78	80.97 ± 0.50	64.15 ± 1.39
X_g+X_y **(ccGAN)**	84.12 ± 0.45	70.68 ± 0.75	80.67 ± 1.29	67.43 ± 0.59	81.40 ± 0.45	$\mathbf{65.20 \pm 1.09}$
X_g	82.67 ± 0.49	67.90 ± 0.80	76.95 ± 1.45	65.20 ± 0.63	74.70 ± 0.80	57.28 ± 1.09

Table 6.4: Predictive performance of RF model in X_g, X_y, and $X_g + X_y$ settings: comparison between our proposed data imputation techniques and other competitors. The best predictive performance result in terms of PRAUC is reported in bold.

Predictors	Accuracy	F1	Precision	Recall	AUC	PRAUC
X_y (KNN)	83.84 ± 0.32	65.70 ± 0.89	89.55 ± 0.86	62.80 ± 0.65	78.12 ± 0.71	59.03 ± 1.09
X_y (MissF)	83.74 ± 0.31	65.80 ± 0.93	88.77 ± 0.68	62.90 ± 0.68	78.76 ± 0.57	60.00 ± 1.21
X_y (MICE)	83.74 ± 0.34	66.30 ± 0.86	87.20 ± 1.20	63.33 ± 0.63	79.54 ± 0.69	60.84 ± 1.21
X_y **(ccGAN)**	83.70 ± 0.44	66.00 ± 1.17	86.60 ± 1.23	63.10 ± 0.86	80.00 ± 0.78	61.60 ± 1.31
X_g+X_y (KNN)	84.23 ± 0.37	68.10 ± 0.84	86.43 ± 1.35	64.75 ± 0.64	81.10 ± 0.82	64.16 ± 1.23
X_g+X_y (MissF)	84.28 ± 0.27	68.21 ± 0.59	86.64 ± 1.06	64.81 ± 0.45	81.38 ± 0.76	64.65 ± 1.03
X_g+X_y (MICE)	84.30 ± 0.39	68.36 ± 0.86	86.47 ± 1.36	64.94 ± 0.65	82.28 ± 0.50	65.53 ± 1.04
X_g+X_y **(ccGAN)**	84.30 ± 0.45	68.60 ± 1.05	86.10 ± 1.29	65.14 ± 0.81	82.67 ± 0.58	$\mathbf{66.16 \pm 1.09}$
X_g	83.91 ± 0.28	67.89 ± 0.70	84.28 ± 0.77	64.66 ± 0.55	78.12 ± 0.54	60.50 ± 1.05

6.5.4 Statistical Analysis

The PRAUC scores in the CV-10 experimental procedure deviate from normality according to the Anderson-Darling test ($p < .01$). Hence, Accordingly, the statistical comparison between our proposed ccGAN data imputation approach and the best data imputation competitors was performed by means of a non-parametric, one-sided Wilcoxon signed-rank test ($\alpha = 0.05$) for the RF and XGB models. The performance of the ccGAN ($X_g + X_y$ setting) is significantly greater ($p < 0.05$) than MICE ($X_g + X_y$ setting) by applying the

RF model. Furthermore, the performance of the ccGAN ($X_g + X_y$ setting) continues to be significantly greater ($p < 0.05$) than MICE ($X_g + X_y$ setting) by applying the XGB model.

6.6 Discussion

We proposed a ccGAN architecture capable to impute missing values from routine EHR data collected from a multi-diabetic centers platform. We demonstrated how the proposed data imputation strategy is consistent for predicting DR in conditions of high missingness rates (i.e. where between 3% and 40% of patients have the candidate feature missing). Within a DR screening programme, our method is currently integrated into a clinical decision support system and permits to discover the most discriminative predictors by also taking into account the missing information.

Our proposed ccGAN data imputation strategy turned out to be robust and effective in dealing with challenging real EHR datasets, characterized by high sparsity, imbalanced setting, noisy and redundant features. This fact motivates that a correct and ad-hoc missing values imputation mechanism could be potentially crucial to obtain a satisfactory predictive performance on routine EHR data.

A limitation of this work might be the exclusion of the X_r predictors (i.e., more than 40% of missing values per patient and more than 80% of missing values for the whole dataset) in the data imputation mechanism. These features, also if selected as important by the diabetologists, were not imputed due to too high missingness rates. Another important limitation might be the exclusion of other EHR fields such as exam and drug prescriptions. These fields may potentially contain a huge amount of missing information related to the availability of a generic drug code (parent code) and the missing of a specific unique drug code (child code). As future work, we aim to exploit a multi-view learning strategy that encapsulates our ccGAN data imputation approach for imputing missing values conditioned to different (eventually missing) views of the EHR dataset and exploiting the structure of a related unlabeled training data [50]. The ccGAN data imputation strategy will also be extended to other diabetes complications by figuring to develop a fully-equipped clinical platform for the management of the diabetic patient.

Chapter 7

Conclusions

Firstly, application-wise, this thesis investigated practical ML solutions for semiconductor manufacturing tasks, like predictive maintenance and virtual metrology. The work highlighted the main challenges in the area of study for the learning algorithms to be used effectively, which are: a level of sensory data pre-processing required for abundant raw equipment sensory data; and a treatment of highly diversified data collections provided for learning; which moreover are collected at a different cost, and therefore, some modalities are not available for all the observations. As a result, we:

- proposed process characterization method that combines clusterization technique employing Gaussian Mixture Models algorithm to perform equipment sensory data segmentation and traditional features extraction technique, allowing to define the informative predictors for the modeling of the natural system state progression and virtual metrology predictions;
- presented experimental results highlighting the benefits of adding design-aware features with in-fab data;
- defined a virtual cross metrology system in which measurements of present and past process steps are incorporated to better characterize the full process sequence.

Secondly, we deeply investigated the last mentioned challenge that is a common problem in many applications besides semiconductor manufacturing tasks – multi-view learning with missing views and :

- proposed a novel tripartite GAN model for applications with two views that makes class prediction along with the generation of missing view in both input spaces in the case when the corresponding modalities are not observed;
- presented experimental results highlighting a state-of-the-art performance compared to multi-view approaches that rely on external view-generating functions, as well as to other GAN-based models that solve missing data imputation problems;
- showcased the proposed model's use in different application domains.

As for future work, the proposed ML solutions for advanced process control tasks are planned to be experimented with other semiconductor processes data, as well as the proposed approaches are planned to be investigated for the use with transfer learning techniques for better generalization purposes. Then, the contributions in the multi-view learning with missing views set a promising direction for further investigation and research to propose a straightforward approach to deal with more than two views in the problem using GAN joint learning strategy as a base.

Bibliography

[1] Umberto Amato, Anestis Antoniadis, and Italia Feis. Flexible, boundary adapted, nonparametric methods for the estimation of univariate piecewise-smooth functions. *Statistics Surveys*, 14:32–70, 01 2020.

[2] Umberto Amato, Anestis Antoniadis, Italia Feis, and Irène Gijbels. Penalized wavelet estimation and robust denoising for irregular spaced data. *Computational Statistics*, 37, 09 2022.

[3] Massih R. Amini, Nicolas Usunier, and Cyril Goutte. Learning from multiple partially observed views - an application to multilingual text categorization. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc., 2009.

[4] Massih-Reza Amini and Cyril Goutte. A co-classification approach to learning from multilingual corpora. *Machine Learning Journal*, 79(1-2):105–121, 2010.

[5] P. C. Andricacos, C. Uzoh, J. O. Dukovic, J. Horkans, and H. Deligianni. Damascene copper electroplating for chip interconnections. *IBM Journal of Research and Development*, 42(5):567–574, 1998.

[6] Francis R. Bach and Michael I. Jordan. Kernel independent component analysis. *Journal of Machine Learning Research*, pages 1–48, 2003.

[7] Francis R. Bach, Gert R. G. Lanckriet, and Michael I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *Proceedings of the* 21^{st} *International Conference on Machine Learning (ICML)*, 2004.

[8] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *ArXiv*, abs/1803.01271, 2018.

[9] M.D. Baker, C.D. Himmel, and G.S. May. Time series modeling of reactive ion etching using neural networks. *IEEE Transactions on Semiconductor Manufacturing*, 8(1):62–71, 1995.

[10] Michele Bernardini, Luca Romeo, Paolo Misericordia, and Emanuele Frontoni. Discovering the type 2 diabetes in electronic health records using the sparse balanced support vector machine. *IEEE Journal of Biomedical and Health Informatics*, 24(1):235–246, 2020.

[11] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory (COLT)*, pages 92–100, 1998.

[12] Ashish Bora, Eric Price, and Alexandros G. Dimakis. AmbientGAN: Generative models from lossy measurements. In *International Conference on Learning Representations*, 2018.

[13] Shane Butler and John Ringwood. Particle filters for remaining useful life estimation of abatement equipment used in semiconductor manufacturing. In *2010 Conference on Control and Fault-Tolerant Systems (SysTol)*, pages 436–441, 2010.

[14] Sujata Butte, Prashanth A R, and Sainath Patil. Machine learning based predictive maintenance strategy: A super learning approach with deep neural networks. *2018 IEEE Workshop on Microelectronics and Electron Devices (WMED)*, pages 1–5, 2018.

[15] Gavin C Cawley and Nicola LC Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11(Jul):2079–2107, 2010.

[16] Yaw-Jen Chang. Wavelet-based virtual metrology technique. In *2010 IEEE International Conference on Mechatronics and Automation*, pages 367–371, 2010.

[17] Chieh-Yu Chen, Shi-Chung Chang, and Da-Yin Liao. Equipment anomaly detection for semiconductor manufacturing by exploiting unsupervised learning from sensory data. *Sensors*, 20(19), 2020.

[18] Chieh-Yu Chen, Shi-Chung Chang, and Da-Yin Liao. Equipment anomaly detection for semiconductor manufacturing by exploiting unsupervised learning from sensory data. *Sensors*, 20(19), 2020.

[19] Mickaël Chen and Ludovic Denoyer. Multi-view Generative Adversarial Networks. In Springer, editor, *ECML PKDD 2017*, volume 10535 of *Machine Learning and Knowledge Discovery in Databases*, pages 175–188, 2017.

[20] PingHsu Chen, S. Wu, Junshien Lin, F. Ko, H. Lo, J. Wang, C.H. Yu, and M.S. Liang. Virtual metrology: a solution for wafer to wafer advanced process control. In *ISSM 2005, IEEE International Symposium on Semiconductor Manufacturing, 2005.*, pages 155–157, 2005.

[21] Fan-Tien Cheng, Jonathan Yung-Cheng Chang, Hsien-Cheng Huang, Chi-An Kao, Ying-Lin Chen, and Ju-Lei Peng. Benefit model of virtual metrology and integrating avm into mes. *IEEE Transactions on Semiconductor Manufacturing*, 24(2):261–272, 2011.

[22] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8789–8797, 2018.

[23] Emily L Denton, Soumith Chintala, arthur szlam, and Rob Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[24] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. In *International Conference on Learning Representations*, 2017.

[25] Paul-Arthur Dreyfus, Foivos Psarommatis, Gökan MAY, and Dimitris Kiritsis. Virtual metrology as an approach for product quality estimation in industry 4.0: a systematic review and integrative conceptual framework. *International Journal of Production Research*, 60, 09 2021.

[26] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Alex Lamb, Martin Arjovsky, Olivier Mastropietro, and Aaron Courville. Adversarially learned inference. In *International Conference on Learning Representations*, 2017.

[27] Pedro Espadinha-Cruz, Radu Godina, and Eduardo M. G. Rodrigues. A review of data mining applications in semiconductor manufacturing. *Processes*, 9(2), 2021.

[28] Jason Farquhar, David Hardoon, Hongying Meng, John Shawe-taylor, and Sándor Szedmák. Two view learning: Svm-2k, theory and practice. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2005.

[29] Lorenz Fedele. From basic maintenance to advanced maintenance. 2011.

[30] A. Ferreira, A. Roussy, and L. Conde. Virtual metrology models for predicting physical measurement in semiconductor manufacturing. In *2009 IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, pages 149–154, 2009.

[31] Konstantinos Fokianos and M Pitsillou. Testing independence for multivariate time series via the auto-distance correlation matrix. *Biometrika*, 105(2):337–352, 2018.

[32] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[33] Anil Goyal, Emilie Morvant, Pascal Germain, and Massih-Reza Amini. PAC-Bayesian Analysis for a two-step Hierarchical Multiview Learning Approach. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 205–221, 2017.

[34] Yu-Ming Hsieh, Tan-Ju Wang, Chin-Yi Lin, Li-Hsuan Peng, Fan-Tien Cheng, and Sui-Yan Shang. Convolutional neural networks for automatic virtual metrology. *IEEE Robotics and Automation Letters*, 6(3):5720–5727, 2021.

[35] Min-Hsiung Hung, Tung-Ho Lin, Fan-Tien Cheng, and Rung-Chuan Lin. A novel virtual metrology scheme for predicting cvd thickness in semiconductor manufacturing. *IEEE/ASME Transactions on Mechatronics*, 12(3):308–316, 2007.

[36] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017.

[37] Aftab A. Khan, James R. Moyne, and Dawn M. Tilbury. An approach for factory-wide control utilizing virtual metrology. *IEEE Transactions on Semiconductor Manufacturing*, 20(4):364–375, 2007.

[38] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML, page 1857–1865, 2017.

[39] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[40] Dongwook Lee, Junyoung Kim, Won-Jin Moon, and Jong Chul Ye. CollaGAN: Collaborative GAN for Missing Image Data Imputation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2487–2496, 2019.

[41] Steven Cheng-Xian Li, Bo Jiang, and Benjamin M. Marlin. MisGAN: Learning from Incomplete Data with Generative Adversarial Networks. *CoRR*, 2019.

[42] Roderick JA Little and Donald B Rubin. *Statistical analysis with missing data*, volume 793. John Wiley & Sons, 2019.

[43] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[44] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[45] Shane Lynn, John Ringwood, and Niall MacGearailt. Gaussian process regression for virtual metrology of plasma etch. In *IET Irish Signals and Systems Conference (ISSC 2010)*, pages 42–47, 2010.

[46] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[47] Marco Maggipinto, Alessandro Beghi, Seán McLoone, and Gian Antonio Susto. Deepvm: A deep learning-based approach with automatic feature extraction for 2d input data virtual metrology. *Journal of Process Control*, 84:24–34, 2019.

[48] Marco Maggipinto, Chiara Masiero, Alessandro Beghi, and Gian Antonio Susto. A convolutional autoencoder approach for feature extraction in virtual metrology. *Procedia Manufacturing*, 17:126–133, 2018. 28th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM2018), June 11-14, 2018, Columbus, OH, USAGlobal Integration of Intelligent Manufacturing and Smart Industry for Good of Humanity.

[49] R Malarvizhi and Antony Selvadoss Thanamani. K-nearest neighbor in missing data imputation. *International Journal of Engineering Research and Development*, 5(1):5–7, 2012.

[50] Yury Maximov, Massih-Reza Amini, and Zaïd Harchaoui. Rademacher complexity bounds for a penalized multi-class semi-supervised algorithm. *Journal of Artificial Intelligence Research*, 61:761–786, 2018.

[51] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier GANs. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 2642–2651, 2017.

[52] Marina Paolanti, Luca Romeo, Andrea Felicetti, Adriano Mancini, Emanuele Frontoni, and Jelena Loncarski. Machine learning approach for predictive maintenance in industry 4.0. In *2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, pages 1–6, 2018.

[53] Pks Prakash, Andrea Schirru, Peter Hung, and Seán McLoone. Msc-clustering and forward stepwise regression for virtual metrology in highly correlated input spaces. In *2012 SEMI Advanced Semiconductor Manufacturing Conference*, pages 45–50, 2012.

[54] Hendrik Purwins, Bernd Barak, Ahmed Nagi, Reiner Engel, Uwe Höckele, Andreas Kyek, Srikanth Cherla, Benjamin Lenz, Günter Pfeifer, and Kurt Weinzierl. Regression methods for virtual metrology of layer thickness in chemical vapor deposition. *IEEE/ASME Transactions on Mechatronics*, 19(1):1–8, 2014.

[55] Sehrish Qummar, Fiaz Gul Khan, Sajid Shah, Ahmad Khan, Shahaboddin Shamshirband, Zia Ur Rehman, Iftikhar Ahmed Khan, and Waqas Jadoon. A deep learning ensemble approach for diabetic retinopathy detection. *IEEE Access*, 7:150530–150539, 2019.

[56] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[57] Yongyi Ran, Xiaoxia Zhou, Pengfeng Lin, Yonggang Wen, and Ruilong Deng. A survey of predictive maintenance: Systems, purposes and approaches. *ArXiv*, abs/1912.07383, 2019.

[58] World Health Organization et al. Diabetic retinopathy screening: a short guide: increase effectiveness, maximize benefits and minimize harm. 2020.

[59] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[60] Vikas Sindhwani and David S. Rosenberg. An RKHS for multi-view learning and manifold co-regularization. In *Proceedings of the* 25^{th} *International Conference on Machine Learning (ICML)*, 2008.

[61] Jost Tobias Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *CoRR*, abs/1511.06390, 2016.

[62] Daniel J Stekhoven and Peter Bühlmann. Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2012.

[63] Gian Antonio Susto, Alessandro Beghi, and Cristina De Luca. A predictive maintenance system for epitaxy processes based on filtering and prediction techniques. *IEEE Transactions on Semiconductor Manufacturing*, 25(4):638–649, 2012.

[64] Gian Antonio Susto, Andrea Schirru, Simone Pampuri, Seán McLoone, and Alessandro Beghi. Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Transactions on Industrial Informatics*, 11(3):812–820, 2015.

[65] Kerul Suthar, Devarshi Shah, Jin Wang, and Q. Peter He. Next-generation virtual metrology for semiconductor manufacturing: A feature-based framework. *Computers Chemical Engineering*, 127:140–149, 2019.

[66] Easter S. Suviseshamuthu, Didier Allexandre, Umberto Amato, Biancamaria Della Vecchia, and Guang H. Yu. Prolific: A fast and robust profile-likelihood-based muscle onset detection in electromyogram using discrete fibonacci search. *IEEE Access*, 8:105362–105375, 2020.

[67] Yaniv Taigman, Adam Polyak, and Lior Wolf. Unsupervised cross-domain image generation. In *International Conference on Learning Representations*, ICLR, 2017.

[68] Lai Tian, Feiping Nie, and Xuelong Li. A unified weight learning paradigm for multi-view learning. In *Proceedings of Machine Learning Research*, pages 2790–2800, 2019.

[69] Yu Tian, Xi Peng, Long Zhao, Shaoting Zhang, and Dimitris N. Metaxas. CR-GAN: Learning Complete Representations for Multi-view Generation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI)*, pages 942–948, 2018.

[70] Nicola Ueffing, Michel Simard, Samuel Larkin, and Howard Johnson. Nrc's portage system for wmt 2007. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 185–188, 2007.

[71] Nicolas Usunier, Massih-Reza Amini, and Cyril Goutte. Multiview semi-supervised learning for ranking multilingual documents. In *European Conference on Machine Learning and Knowledge Discovery in Databases (ECMLPKDD)*, pages 443–458, 2011.

[72] Marcela Vallejo, Carolina de la Espriella, Juliana Gómez-Santamaría, Andrés Ramírez, and Edilson Delgado-Trejos. Soft metrology based on machine learning: A review. *Measurement Science and Technology*, 31, 10 2019.

[73] Jian Wan, Simone Pampuri, Paul G. O'Hara, Adrian B. Johnston, and Seán McLoone. On regression methods for virtual metrology in semiconductor manufacturing. In *25th IET Irish Signals Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014)*, pages 380–385, 2014.

[74] Chang Xu, Dacheng Tao, and Chao Xu. A survey on multi-view learning. *CoRR*, abs/1304.5634, 2013.

[75] Z. Yi, H. Zhang, P. Tan, and M. Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2868–2876, Los Alamitos, CA, USA, 2017. IEEE Computer Society.

[76] Jinsung Yoon, James Jordon, and Mihaela Schaar. Gain: Missing data imputation using generative adversarial nets. In *International Conference on Machine Learning*, pages 5689–5698. PMLR, 2018.

[77] Han Qiu Zhang and Yong Yan. A wavelet-based approach to abrupt fault detection and diagnosis of sensors. *IEEE Transactions on Instrumentation and Measurement*, 50(5):1389–1396, 2001.

[78] Rui Zhao, Ruqiang Yan, Zhenghua Chen, Kezhi Mao, Peng Wang, and Robert X. Gao. Deep learning and its applications to machine health monitoring. *Mechanical Systems and Signal Processing*, 115:213–237, 2019.

[79] Zhou Zhou. Measuring nonlinear dependence in time-series, a distance correlation approach. *Journal of Time Series Analysis*, 33(3):438–457, 2012.

[80] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017.