Journal of Intelligent Manufacturing (2025) 36:357–372
https://doi.org/10.1007/s10845-023-02245-7

Natural language processing (NLP) and association rules (AR)-based knowledge extraction for intelligent fault analysis: a case study in semiconductor industry

Zhiqiang Wang[1] · Kenneth Ezukwoke[2,3] · Anis Hoayek[2,3] · Mireille Batton-Hubert[2,3] · Xavier Boucher[2,4]

Received: 29 June 2023 / Accepted: 10 October 2023 / Published online: 9 November 2023

Abstract

Fault analysis (FA) is the process of collecting and analyzing data to determine the cause of a failure. It plays an important role in ensuring the quality in manufacturing process. Traditional FA techniques are time-consuming and labor-intensive, relying heavily on human expertise and the availability of failure inspection equipment. In semiconductor industry, a large amount of FA reports are generated by experts to record the fault descriptions, fault analysis path and fault root causes. With the development of Artificial Intelligence, it is possible to automate the industrial FA process while extracting expert knowledge from the vast FA report data. The goal of this research is to develop a complete expert knowledge extraction pipeline for FA in semiconductor industry based on advanced Natural Language Processing and Machine Learning. Our research aims at automatically predicting the fault root cause based on the fault descriptions. First, the text data from the FA reports are transformed into numerical data using Sentence Transformer embedding. The numerical data are converted into latent spaces using Generalized-Controllable Variational AutoEncoder. Then, the latent spaces are classified by Gaussian Mixture Model. Finally, Association Rules are applied to establish the relationship between the labels in the latent space of the fault descriptions and that of the fault root cause. The proposed algorithm has been evaluated with real data of semiconductor industry collected over three years. The average correctness of the predicted label achieves 97.8%. The method can effectively reduce the time of failure identification and the cost during the inspection stage.

Keywords Fault analysis · Natural language processing · GCVAE · GMM · Association rules

Introduction

Research motivation

Fault analysis (FA) is the process of determining the root cause of a failure, collecting and analyzing data, as well as drawing conclusions to eliminate the Failure Mechanism (FM) (Martin, 1999). It is primarily conducted in safety-critical applications, such as automotive, aerospace, marine, semiconductor, and digital systems. For crucial electronic components and systems, FA is one of the most important steps in reliability analysis (Bajenescu & Bazu, 2012).

Kenneth Ezukwoke, Anis Hoayek, Mireille Batton Hubert and Xavier Boucher have contributed equally to this work.

✉ Zhiqiang Wang
zhiqiang.wang@devinci.fr

Kenneth Ezukwoke
ifeanyi.ezukwoke@emse.fr

Anis Hoayek
anis.hoayek@emse.fr

Mireille Batton-Hubert
batton@emse.fr

Xavier Boucher
boucher@emse.fr

[1] Léonard de Vinci Pôle Universitaire, Research Center, 92 916 Paris La Défense, France

[2] Mines Saint-Étienne, Univ. Clermont Auvergne, CNRS UMR 6158 LIMOS, 42100 Saint-Étienne, France

[3] Mathematics and Industrial Engineering, Henri FAYOL Institute, 42023 Saint-Étienne, France

[4] Center for Biomedical and Healthcare Engineering, 42023 Saint-Étienne, France

In the field of microelectronics, the root causes of the fault can come from different aspects: manufacturing process, design, environment, and maintenance. To figure out the root cause, traditional FA techniques include visual examination, physical or chemical analysis, nondestructive testing, package construction analysis, fault localization, circuit edit, wafer construction analysis, etc. On the one hand, the majority of these traditional FA techniques are strongly dependent on human expertise by involving manual analysis and inspections, which are often time and labor-consuming. On the other hand, a vast amount of data is generated during the whole FA procedure. The data includes not only numeric and image information recorded during the inspections but also textual documents provided by experts reporting their tasks, findings, and conclusions regarding each device (Platter et al., 2021).

To overcome the limitations of the traditional FA techniques, the big data generated during the traditional FA needs to be deeply explored. Artificial intelligence (AI) is opening a new era to help FA explore this valuable data. The main objective of this paper is to propose a Natural Language Processing (NLP) and Association Rule (AR)-based method to efficiently extract expert knowledge of FA applied on semiconductor industry. Here, the proposed FA algorithm belongs to the Safety-I approach defined in Hollnagel (2018). For Safety-I approach, failure events are taken as the focus point and the overall method tries to prevent failures from occurring, while the Safety-II approach focuses on non-failure cases.

Context and main contributions of this paper

To benefit from the historical FA reports, this article proposes a complete AI-based knowledge extraction pipeline to automatically predict the final failure cause based on the failure description. To this end, NLP-based techniques are first applied on a huge industrial dataset provided by STMicroelectronics to convert text data into numerical data as well as to model the data in a latent space with reduced dimension. In our previous study (Wang et al., 2022), Word2vect was used to embed the text data then Gaussian Mixture Model (GMM) was applied to perform clustering in the latent space of the Fault Root Cause Space (FRCS) obtained by the Variational AutoEncoder (VAE). FRCS is a space that summarizes the fault root cause recorded in the FA report through the Fault Reporting, Analysis, and Corrective Action Systems (FRACAS) (Ezukwoke et al., 2021). Word2vect is proposed by Mikolov et al. (2013) to convert text into numerical data through learning the similarity between tokenized words from different documents with a neural network. As a statistical tool, GMM is proposed in Reynolds et al. (2009) to model the distribution of random variables by a mixture of Gaussian distributions in order to parametrically estimate them. VAE is a type of autoencoder to reduce the dimension of the features by sampling the input data set x_i from a prior distribution $p_\theta(z)$ to the output data x_i' while keeping the output as similar as the input (Kingma & Welling, 2013).

This paper extends our previous research by first applying a more efficient NLP embedding technique, i.e., Sentence Transformer, separately on Fault Description Space (FDS) and Fault Root Cause Space (FRCS). Then, more importantly, Association Rules (AR) are established between the latent space of FDS and FRCS, which link the observed pattern description with the fault conclusion. In this way, a complete close loop knowledge extraction pipeline is developed. The proposed algorithm is able to determine the conclusions of the analysis and find the root cause of the fault more easily, accurately and automatically when the same category of fault occurs. Moreover, the parameters of the model in the proposed method are automatically updated in a continuous way along with new data. This allows the model to be generalized to new fault cases and provides the possibility of continuous improvement in the expert knowledge transfer. To summarize, the main contributions of this work are as follows:

- The text data from the FDS and the FRCS are separately converted into numerical data by NLP using Sentence Transformer;
- The Generalized-Controllable Variational AutoEncoder (GCVAE) is employed to re-represent the embedding space and reduce the original dimension using Bayesian inference to obtain high disentanglement performance and minimal information ;
- After clustering by GMM in the latent spaces of FDS and FRCS, Association Rules (AR) are established to determine the relationship between different categories in FDS and different groups in FRCS.
- By combining the previously proposed modules, a complete AI-based knowledge extraction pipeline is obtained for FA. The method is applied and evaluated on real industrial data of semiconductor domain with a high potential for real-scenario implementation.

The obtained AI model can automatically predict the fault root cause based on the fault description. It can help experts narrow down the potential causes of error before going through different fault analysis inspections. The method can effectively reduce the time of failure identification as well as the cost during the inspection stage.

Organization of the paper

The remainder of this paper is organized as follows: the state-of-the-art technologies regarding AI-based FA are summarized in "State of the art", the proposed method is presented in

"Description of the method". "Experiments results" demonstrates the performances of the proposed method applied on real industrial data. Finally, "Conclusions and perspectives" draws conclusions and proposes future works.

State of the art

In the current literature, much research focuses on the application of AI to improve FA inspections using the numeric data or image data. A multiple time-series convolutional neural network (MTS-CNN) model using the equipment sensors data is proposed by Hsu and Liu (2021) for fault detection and diagnosis in semiconductor manufacturing. The experimental results show that this model outperforms other existing multivariate time-series methods. In Gu et al. (2022), a method of Deep-Learning for High-Resolution Reconstruction (DLHRR) is proposed to improve the scanning speed in 3D non-destructive X-ray microscopy (XRM) for fault analysis. An Automatic Defect Classification system (ADC) using Deep Learning to automatically classify wafer surface defects is proposed by Phua and Theng (2020). Large-scale integration (LSI) layout images were classified using Convolutional neural networks (CNNs) to perform Root Cause Analysis (RCA) of layout-related defects (Nagamura et al., 2021). A novel Hypergraph Convolution Network is proposed to classify the wafer defect images in Xie et al. (2022). However, few researches deal with knowledge extraction from the historical reports of FA which are mostly textual data. The following subsections make a literature review while emphasizing AI-based FA, especially with textual data.

NLP applied in fault analysis

Since most of the data consists of text, Natural Language Processing (NLP) is required to preprocess this textual data first. The efficiency of pre-trained Language Models (LM) in the semiconductor domain for text classification with deep neural networks is studied by Grabner et al. (2022), whose result is not as good as that of the Word2vect model and the Linear Support Vector Classifier (SVC). Trappey et al. (2021) develops the methodology for a patent recommender in smart machinery technology mining of intelligent machines to discover semantically relevant patents for further technology mining and trend analysis. In Ezukwoke et al. (2021), NLP techniques are presented to find a coherent representation of expert decisions in fault analysis in the semiconductor industry. Once the textual fault analysis data is transformed into numerical data by NLP, deep learning technology is applied to both reduce the dimension of this numerical data and cluster this latent space into different groups.

Deep learning in fault analysis

Deep learning based on artificial neural networks can help the industry find the root cause automatically and quickly. In Dimitriou et al. (2019), a system is proposed that automates fault diagnosis by accurately estimating the volume of glue deposits using a three-dimensional (3-D) convolutional neural network (3DCNN) before and even after die attachment. Watanabe et al. (2019) presents image diagnosis by Convolutional Neural Network (CNNs) based image diagnosis applied to power device fault analysis. Wang et al. (2022) compares different deep learning methods based on VAE (Variational AutoEncoder) and concludes that Generalized-Controllable Variational AutoEncoder (GCVAE) by Ezukwoke et al. (2022) is the best model to find an intelligent optimal representation of fault analysis written in natural language. Once we reduce the dimension using VAE-based Deep Learning, we obtain the latent spaces. The machine learning methods are necessary to perform the clustering in these latent spaces. The goal is to cluster the FDS and the FRCS into different groups.

Machine learning in fault analysis

Machine learning is permeating more and more academic disciplines and industries, especially the fields of reliability engineering and safety (Xu & Saleh, 2021). Six machine learning models (Naıve Bayes, decision trees, K-nearest neighbors, quadratic discriminant analysis, random forests, and artificial neural networks) are applied and compared to predict the failure mode in circular reinforced bridge piers in Mangalathu and Jeon (2019). The author of the above conclusion that Artificial Neural Network (ANN), outperforms other machine learning models. Unsupervised machine learning is preferred in most industrial applications due to the additional and difficult work for the industry lab in collecting the labeled output data. Wang et al. (2020) compared the K-means and Gaussian Mixture Model (GMM) in clustering the machining data in real-time and concluded that GMM performs better on unbalanced data, which is exactly applicable to our case. The distribution of latent space obtained by GCVAE in the above section could be modeled as a mixture of Gaussian distributions (different clusters), which is consistent with the concept of GMM. The larger the number of parameters to be estimated in the GMM model, the better the model is able to generalize the data. However, the complexity of the calculations also increases. Therefore, Bayesian information criterion-BIC (Schwarz, 1978) is used to find a trade-off between the optimal number of clusters in the GMM and the complexity of the calculations.

After obtaining the different groups in FDS and FRCS by clustering, we need to find the potential correlation between different groups in FDS and different groups in FRCS. To achieve this, Association rule learning is preferred in this case. The Association Rules (AR) proposed by Agrawal et al. (1993) aim to discover the rules that determine the potential association between elements in each transaction with a large number of elements. Antomarioni et al. (2022) proposes a framework that aims to deepen the fault analysis by applying association rule mining and social network analysis. In Antonello et al. (2021), the association rule mining algorithm is applied to identify groups of functionally dependent components in a complex technical infrastructure. The most common evaluation metrics for AR are support, confidence, and lift (Hahsler et al., 2005). They are the metrics used to establish the intercorrelation between the different groups in FDS and FRCS.

Description of the method

Dataset structure description

The dataset used in this paper is a real industrial one provided by STMicroelectronics through the Fault Reporting, Analysis, and Corrective Action System (FRACAS) from 2019 to 2021 (Ezukwoke et al., 2021). It consists of 12,032 observations and 134 features, where 88 features consist of text or categories and 46 features consist of numbers. Considering that the majority of the data is in text format, the application of NLP techniques is required. Three different sections could be found in this data set:

- Fault Description Space (FDS) which describes the fault context or reference (denoted x_n);
- Analysis Paths Space (APS) which represents the fault analysis triplets registered by the industry (denoted λ_n);
- Fault Root Cause Space (FRCS) which describes the failure cause (denoted y_n).

Figure 1 illustrates the FA decision flow graphic followed by industry with FDS, APS and FRCS. Different failure descriptions $\{A, B, C, D \ldots\}$ have different observation length $n \in [1, i-1]$.

The FDS includes attributes such as *Reference*, *Context*, *Subject*, *Requestor*, etc. Table 1 shows all the 25 features in FDS as well as their descriptions and data examples. The analysis paths (λ_n) are composed of FA Triplets (FATs) which is a series of *Step type*, *Substep technique*, and *Equipment* proposed by a failure analyst regarding the actions to find the failure root cause. For example, 'Non-destructive Inspection' is the *Step type* and 'X-ray' is the *Substep technique* while '3D X-RAY' is the name of *Equipment*. The number of these actions is different for each failure description so a padding is made based on the longest FAT. Finally, the FRCS (y_n) is represented by *Analysis conclusion*, *Tech cause / Defect by sample*, etc. as shown in Table 2. In this work, the objective is to predict the probable cause of the defect by determining the relationships between the FDS and the FRCS through NLP, machine learning and deep learning.

Overview of the method

Figure 2 presents an overview of the proposed NLP-based knowledge extraction architecture for FA using the natural language database of historical fault analysis reports. The green dashed area represents the work in the previous study, while the purple dashed area illustrates the contributions in this article. Most of the features in this report are text data, an NLP method is required to preprocess the FDS data and the FRCS data. Then, the text data is converted into numerical data using the Sentence Transformer proposed in Reimers and Gurevych (2019). The numeric dataset contains many features (384 for FDS and 384 for FRCS). To reduce the dimension of this large dataset containing complex nonlinear features, Auto-Encoders (AE) are preferred since they can model complex nonlinear functions (Hinton & Salakhutdinov, 2006). We also need to perform clustering in the latent space, which can be achieved by Variational AutoEncoders (VAE). After that, the latent spaces with two dimensions for FDS and FRCS are respectively obtained. The unsupervised machine learning based on Gaussian Mixture Model (GMM) is used to cluster these two latent spaces. Then, the labels in the FDS and FRCS are determined. Finally, the AR are established to find the potential relationship between the labels in FDS and in the FRCS. The details of each module will be presented in the following sub-section.

Data preprocessing pipeline and vectorization

Given the set of variables contained in FRCS ($\{y_i\}_{i=0}^{n} \in \mathbb{R}^D$), where D is the dimension of y, are textual, and contain exactly 7 features (Table 2). Similarly, the data type of the input variables for the given FDS ($\{x_i\}_{i=0}^{n} \in \mathbb{R}^D$) are textual variables and, in particular, include 25 features (Table 1). We need to preprocess the FDS and the FRCS independently and identically (as seen in Fig. 2) using the following preprocessing pipeline, which is divided into five main steps. The first four steps of preprocessing are the same as in our previous work (Wang et al., 2022). In addition, the Sentence Transformer is applied to convert the textual data into numerical data, as it's able to derive semantically meaningful sentence embeddings (Reimers & Gurevych, 2019). The five main preprocessing steps applied to the FDS and FRCS variables are as follows:

- Cleaning and preprocessing: involves removing unwanted alphanumeric words and symbols from the text that do

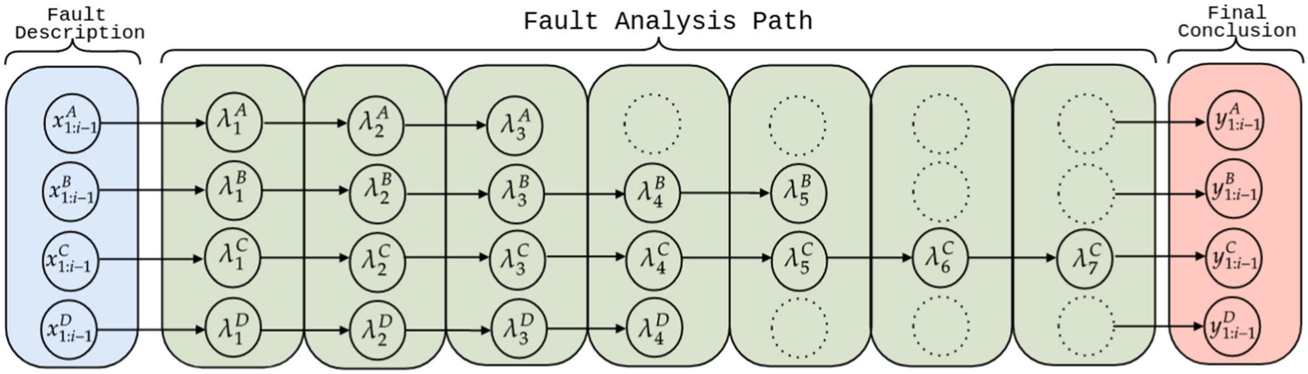

Fault Description
Fault Analysis Path
Final Conclusion
$x^A_{1:i-1}$
$x^B_{1:i-1}$
$x^C_{1:i-1}$
$x^D_{1:i-1}$
λ^A_1 λ^A_2 λ^A_3
λ^B_1 λ^B_2 λ^B_3 λ^B_4 λ^B_5
λ^C_1 λ^C_2 λ^C_3 λ^C_4 λ^C_5 λ^C_6 λ^C_7
λ^D_1 λ^D_2 λ^D_3 λ^D_4
$y^A_{1:i-1}$
$y^B_{1:i-1}$
$y^C_{1:i-1}$
$y^D_{1:i-1}$

Fig. 1 FA decision flow graphic: Fault Description Space, Fault Analysis Path Space and Fault Root Cause Space. A, B, C, D are different failure descriptions with different numbers of observations n

Table 1 Brief description of the failure description space (FDS) features and their corresponding data examples

Features in FDS	Description	Data example
Reference	Associated the fault analysis team, FA-lab, year and ID	AGR_DMA_18_00115
Subject	A particular unique subject for the fault expert desire to analyze	H9A load failure
Context	Context of the fault analysis	Wafer from SINGAPORE IDDq reject SG9 767
Objectives/work description	Objective of the fault analysis	Check for delamination
Source of failure/request	Identifier source of the failure for the component of a given sample	Reliability/monitoring
Source of failure/(detailed)	Details of the source of the failure	Manufacturing
Organization	The organization handling FA	DMA
Organization/division	The organization division handling FA	Digital quality
Department	FA department handling FA	Indust process
Requestor	The requestor requesting for FA of the sample	GODUCHEAU Olivier
Cost center	Identifier corresponding to the cost of FA for a sample	AG6150
Confidentiality	Level of confidentiality of FA	ST only
Site	Fault analysis lab site	Grenoble
Lab	The section of the division where analysis or sub-analysis is done	Cornaredo FA-Lab
Lab team	Lab division name	AGR_DMA
Requested activity	Major fault analysis requested by expert or client	Fault analysis
High confidentiality	Unique identifier stating if FA is confidential high or not	Yes/no
Project	Unique reference number assigned to an FA	DMA_Digital
Priority level	Level of priority given to failure device	P0, P1, P2 or P3
Date of creation	Date of creation of the fault analysis record	02-JAN-19 16:55
Date of validation	Validation date to begin fault analysis	26-MAR-19 14:47
Requestor expected date	Date expected by the requestor to begin fault analysis	28-FEB-19
Lab team forecast date	Forecast date to begin fault analysis	28-FEB-19
Starting date of request	Date of start of a request	03-JAN-19 07:10
Last transition date	Last date of transition of the sample	05-JAN-19

not contribute to the analysis. This includes the removal of stop words and inflections;

- Text Tokenization: Tokenization involves breaking down texts into their smallest units, whereupon a threshold is applied to remove words of short length. In our case, words below a length of three are removed for all input/output;
- Stemming and Lemmatization: removal of suffixes and inflections to transform words into their original base form;
- Abbreviation: Because abbreviations are common in fault analysis reports, an abbreviation dictionary is used to compare and replace abbreviations with their original meaning;

Table 2 Brief description of the Fault Root Cause Space (FRCS) features and their corresponding examples

Features in FRCS	Description	Data example
Analysis conclusion	Final conclusion of the analysis by experts	Random wafer fab defect under Cu RDL metal level
Global success rate	Success rate of the overall fault analysis	Successful
Unsuccessful reason	Reasons for the failure of an analysis to yield expected result	No physical defect found
Macro failure mode by sample	Macro failure mode	Parametric failure
Pt failure/Elt by sample	Location of defect	BEoL_Die edge/wafer sawing street
Elementary failure mode	Elementary failure observed during analysis	Short
Tech cause/defect by sample	Technical cause of the failure	Large silicon melting

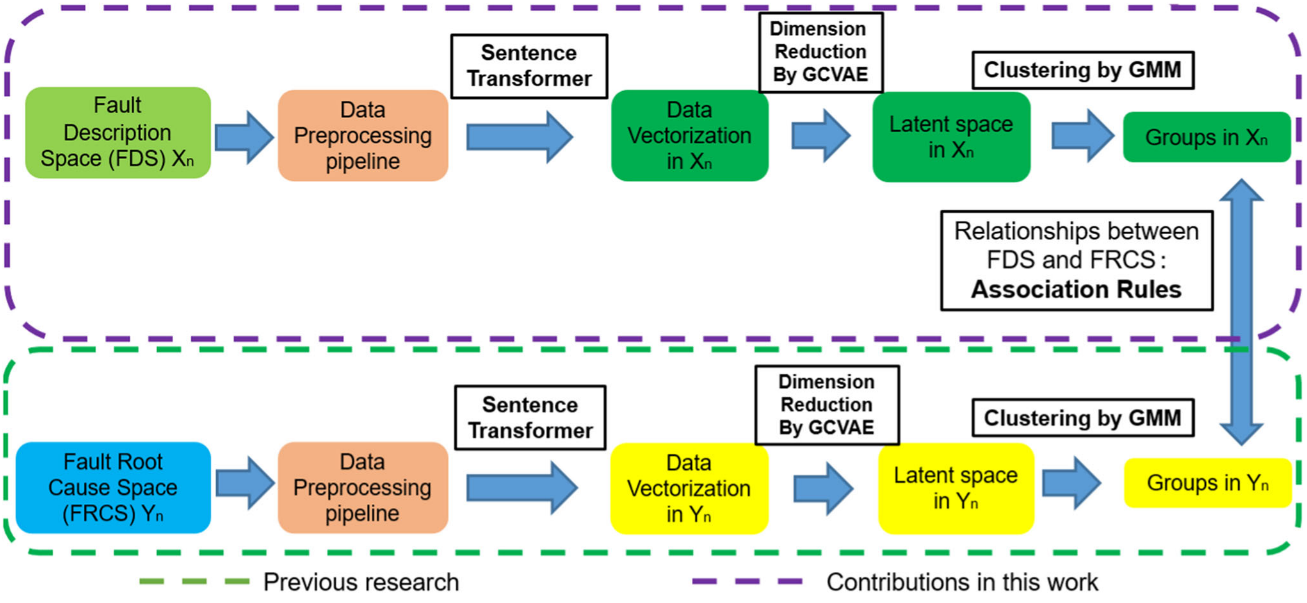

Fault Description Space (FDS) Xn
Data Preprocessing pipeline
Sentence Transformer
Data Vectorization in Xn
Dimension Reduction By GCVAE
Latent space in Xn
Clustering by GMM
Groups in Xn
Relationships between FDS and FRCS : Association Rules
Fault Root Cause Space (FRCS) Yn
Data Preprocessing pipeline
Sentence Transformer
Data Vectorization in Yn
Dimension Reduction By GCVAE
Latent space in Yn
Clustering by GMM
Groups in Yn
Previous research
Contributions in this work

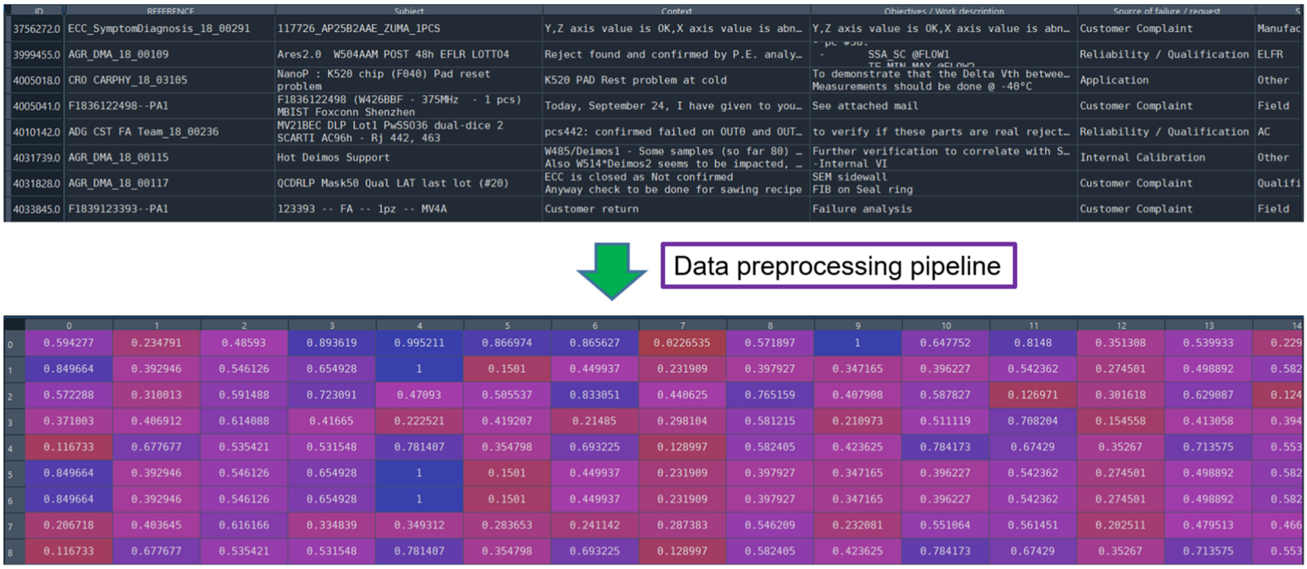
Fig. 2 An overview of the proposed NLP-based knowledge extraction architecture for FA: the green dashed area represents the previous study and the purple dashed area illustrates the contributions in this work (Color figure online)

- Sentence Transformer: Sentence Transformer proposed by Reimers and Gurevych (2019) is an efficient NLP technique for converting text into numerical data. In this algorithm, semantically similar sentences can be found by the similarity measure such as cosine similarity or Manhattan/Euclidean distance from different documents using a neural network. In this work, all features are transformed separately in the FDS and FRCS. Therefore, the text data is transformed into numerical data.

After Sentence Transformer, we also perform the **'MinMaxScaler'** to keep all of the features inside the range of 0 and 1. Figure 3 represents some examples of features in FDS x_n before data preprocessing and the corresponding features after transformation.

Dimension reduction by VAE

The text data were converted to numeric data by the data preprocessing pipeline. Feature engineering is necessary to reduce the dimension of numerical data because a large amount of features are generated (384 features in FRCS). Auto-Encoders (AE) perform better than Principal Component Analysis (PCA) on complex nonlinear data in reducing the dimension of numerical data (Hinton & Salakhutdinov, 2006), which is also the case in our work. Moreover, Variational AutoEncoders (VAE) are used to reduce the data dimension and reconstruct our original text data, as the latent space generated by GCVAE (Ezukwoke et al., 2022) could help us in clustering. In the article Wang et al. (2022), differ-



ID	REFERENCE	Subject	Context	Objectives / Work description	Source of failure / request	S
3756272.0	ECC_SymptomDiagnosis_18_00291	117726_AP25B2AAE_ZUMA_1PCS	Y,Z axis value is OK,X axis value is abn…	Y,Z axis value is OK,X axis value is abn…	Customer Complaint	Manufac
3999455.0	AGR_DMA_18_00109	Ares2.0 W504AAM POST 48h EFLR LOTT04	Reject found and confirmed by P.E. analy…	- pc #30: - SSA_SC @FLOW1 TF MIN MAX @FLOW2	Reliability / Qualification	ELFR
4005018.0	CRO CARPHY_18_03105	NanoP : K520 chip (F040) Pad reset problem	K520 PAD Rest problem at cold	To demonstrate that the Delta Vth betwee… Measurements should be done @ -40°C	Application	Other
4005041.0	F1836122498--PA1	F1836122498 (W426BBF - 375MHz - 1 pcs) MBIST Foxconn Shenzhen	Today, September 24, I have given to you…	See attached mail	Customer Complaint	Field
4010142.0	ADG CST FA Team_18_00236	MV21BEC DLP Lot1 PwSS036 dual-dice 2 SCARTI AC96h - Rj 442, 463	pcs442: confirmed failed on OUT0 and OUT…	to verify if these parts are real reject…	Reliability / Qualification	AC
4031739.0	AGR_DMA_18_00115	Hot Deimos Support	W485/Deimos1 - Some samples (so far 80) … Also W514*Deimos2 seems to be impacted, …	Further verification to correlate with S… -Internal VI	Internal Calibration	Other
4031828.0	AGR_DMA_18_00117	QCDRLP Mask50 Qual LAT last lot (#20)	ECC is closed as Not confirmed Anyway check to be done for sawing recipe	SEM sidewall FIB on Seal ring	Customer Complaint	Qualifi
4033845.0	F1839123393--PA1	123393 -- FA -- 1pz -- MV4A	Customer return	Failure analysis	Customer Complaint	Field

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	0.594277	0.234791	0.48593	0.893619	0.995211	0.866974	0.865627	0.0226535	0.571897	1	0.647752	0.8148	0.351308	0.539933	0.229
1	0.849664	0.392946	0.546126	0.654928	1	0.1501	0.449937	0.231909	0.397927	0.347165	0.396227	0.542362	0.274501	0.498892	0.582
2	0.572288	0.310013	0.591488	0.723091	0.47093	0.505537	0.833051	0.440625	0.765159	0.407908	0.587827	0.126971	0.301618	0.629087	0.124
3	0.371003	0.406912	0.614088	0.41665	0.222521	0.419207	0.21485	0.298104	0.581215	0.210973	0.511119	0.708204	0.154558	0.413058	0.394
4	0.116733	0.677677	0.535421	0.531548	0.781407	0.354798	0.693225	0.128997	0.582405	0.423625	0.784173	0.67429	0.35267	0.713575	0.553
5	0.849664	0.392946	0.546126	0.654928	1	0.1501	0.449937	0.231909	0.397927	0.347165	0.396227	0.542362	0.274501	0.498892	0.582
6	0.849664	0.392946	0.546126	0.654928	1	0.1501	0.449937	0.231909	0.397927	0.347165	0.396227	0.542362	0.274501	0.498892	0.582
7	0.206718	0.403645	0.616166	0.334839	0.349312	0.283653	0.241142	0.287383	0.546209	0.232081	0.551064	0.561451	0.202511	0.479513	0.466
8	0.116733	0.677677	0.535421	0.531548	0.781407	0.354798	0.693225	0.128997	0.582405	0.423625	0.784173	0.67429	0.35267	0.713575	0.553

Fig. 3 The features in FDS x_n before data preprocessing pipeline and after transformation

ent variational models were compared by applying them to the FRCS (y_n) to reduce the dimension of numerical data, and their performances were evaluated using existing metrics. The Generalised-Controllable VAE (GCVAE) is the most suitable for our case. Subsequently, GCVAE is also applied to reduce the numerical data obtained in the latent space from FDS (x_n).

Clustering in the latent space

Once we reduce the dimension of the latent space to 2 in FDS and FRCS separately, we need to cluster these latent spaces with 2 dimensions. In this work, we use unsupervised machine learning because it is difficult to collect labeled output data in the industry. And the Gaussian Mixture Model (GMM) is most effective for unbalanced data, which is the case in industry. The optimal number of clusters in the GMM is determined by a Bayesian Information Criterion (BIC) (Schwarz, 1978).

Establish the relationship between FDS and FRCS

One main challenge of AI-based knowledge extraction technique is its interpretability especially when the text data is converted into numerical latent space. The question here is how to establish the relationship between the clustered latent spaces of FDS and that of FRCS in order to link the fault description with the fault root cause. To this end, we propose to establish the mapping functions using Association Rules (AR). AR is a data mining technique to discover potential dependencies or rules among different items based on their co-occurrence in events or transactions. They are useful in a variety of domains including marketing, recommendation and decision-making systems.

Let $I = \{i_1, i_2, \ldots, i_n\}$ be a set of n attributes called items. In our case, I is composed of different labels in FDS (x_n) as well as the different labels in FRCS (y_n). Each transaction (or called observation) in our database contains a label of FDS (X_i) and a label of FRCS (Y_j). The objective is to propose the possible fault root cause conclusion (y_n) based on the fault description (x_n) by creating an intercorrelation between each label of FDS (X_i) and each label of FRCS (Y_j).

To ensure a highly meaningful result, the AR need to be established according to several metrics. The most common evaluation indicators for AR are support, confidence and lift (Hahsler et al., 2005).

Support is defined as the occurrence frequency of the itemset associated by a rule in the dataset (Agrawal et al., 1993). In our case, *Support* is defined for each associated labels X_i in FDS and Y_j in FRCS as follow:

$$Support(X\_i \Rightarrow Y\_j) = \frac{\#\{X\_i \cup Y\_j\}}{\#\{T\}} \tag{1}$$

where the notation "#" means the occurrence of an event; $\#\{X\_i \cup Y\_j\}$ is the number of fault events containing observation X_i and fault root cause Y_j, while $\#\{T\}$ is the total number of fault events in our dataset.

Confidence is a metric to indicate how often the rule has been found to be true. In our case, for a rule $X\_i \Rightarrow Y\_j$, it is calculated with the percentage of all fault cases containing the observations X_i that lead to fault root case Y_j among all the fault cases containing observation X_i, as shown in the following equation:

$$Conf(X\_i \Rightarrow Y\_j) = \frac{Support(X\_i \cup Y\_j)}{Support(X\_i)} \quad (2)$$

Lift is an indicator to determine whether the two items are independent. It is the ratio between the observed support and the expected support. It is defined by the Eq. 3. The value of $Lift(X\_i \Rightarrow Y\_j)$ near to 1 indicates X_i and Y_j almost often appear together as expected. If it is greater than 1, that means they appear together more than expected and vice versa. Higher lift values indicate a stronger association:

$$Lift(X\_i \Rightarrow Y\_j) = \frac{Support(X\_i \cup Y\_j)}{Support(X\_i) \cdot Support(Y\_j)}. \quad (3)$$

In summary, the higher these three indicators (*Support*, *Confidence*, *Lift*) are, the better the prediction will be.

Many well-known algorithms for generating AR are proposed in the literature (e.g., Apriori, Eclat in Xiao et al., 2016, and FP-Growth in Ji & Deng, 2007). However, all algorithms require to fix hard thresholds for each indicator (*Support*, *Confidence*, *Lift*). Obtaining these thresholds needs additional industrial expert knowledge and they cannot flexibly adapt to different fault events.

In our case, for each cluster X_i in FDS, all possible association rules are generated with each cluster Y_j in FRCS. The three previously mentioned metrics are calculated and sorted. For each metric, the 5 highest values are selected. In particular, the *Lift* values need to be higher than 1. The final selected rules by each metric are then considered together to provide the final association rules to link fault descriptions and fault root causes. The pseudocode of the proposed algorithm is shown in Algorithm 1 below which is described as follows:

1. Find the cluster to which this fault description belongs according to the clustering in the FDS (x_n), for example, X_i.
2. For each X_i, compute the $Support(X\_i \Rightarrow Y\_j)$, the $Conf(X\_i \Rightarrow Y\_j)$ and the $Lift(X\_i \Rightarrow Y\_j)$ for each cluster Y_j in FRCS (y_n).
3. Keep the rules $X\_i \Rightarrow Y\_j$ for which the $Lift(X\_i \Rightarrow Y\_j)$ is greater than 1. Sort the values for each indicator, and keep the 5 maximum values and the corresponding rules.
4. Till now, each indicator has 5 rules with maximal values. We select the rules that appear in all these three indicators. These rules are proposed to predict the conclusions of the fault analysis.

Algorithm 1 Establishing association rules between the latent spaces of FDS and FRCS

```
for $X\_i \in FDS, \quad i = 1, 2, \ldots, N$ do
    for $Y\_j \in FRCS, \quad j = 1, 2, \ldots, M$ do
        Calculate $Support(X\_i \Rightarrow Y\_j)$
        Calculate $Confidence(X\_i \Rightarrow Y\_j)$
        Calculate $Lift(X\_i \Rightarrow Y\_j)$
    end for
end for

Sort each metric in descending order
For Support and Confidence, keep the rules with the 5 highest values
For Lift, keep the rules with 5 highest values while >1
The intersections of the rules selected according to each metric will
    be the final association rules to link the FDS and FRCS
```

Experiments results

Experimental data

The original data in this paper, taken from the historical fault analysis reports, are provided by STMicroelectronics for the period 2019–2021 and have dimension $\mathbb{R}^{12032 \times 134}$. The original data can be mapped into three spaces as mentioned in "Description of the method". The goal is to help the industry expert by predicting the conclusion of the analysis in FRCS (y_n) based on the FDS (x_n) without or with less help from APS (λ_n). As described in the above section, we need to preprocess the original data, which is mostly in text form.

Preprocessing pipeline

The original data is preprocessed according to the data preprocessing pipeline as explained in "Description of the method". The original data is transferred to the steps of data cleaning, tokenization and thresholding, stemming and lemmatization, and abbreviation. In the last step, the text data are converted to numeric by Sentence Transformer (Reimers & Gurevych, 2019) using semantic analysis of sentences within the algorithm, which is better than Word Embedding in Wang et al. (2022) Then the text data is converted into numeric data with many features. Then the feature engineering method is needed to reduce the dimension of our data.

Features' dimension reducing by GCVAE

As described in the previous research, VAE is chosen because Auto-Encoder (AE) is better suited for complex nonlinear functions (Hinton & Salakhutdinov, 2006), which is our case. Also, we need to perform clustering in latent space, which is only possible with VAE. Many VAE algorithms are applied in the FRCS to reduce the dimension of features in Wang

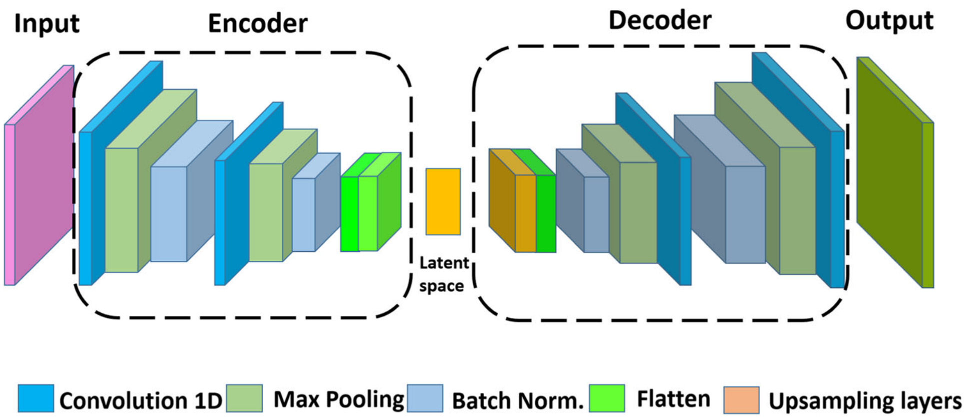


Fig. 4 Encoder–Decoder architecture for GCVAE-II (with MAH distance)

et al. (2022), and GCVAE outperforms the others. Meanwhile, GCVAE-II is chosen (with the expected MaHalanobis Distance-MAH between the density function of two continuous variables) to reduce the dimension of the features in FDS and FRCS separately. Because GCVAE-II can give a better result and the computational complexity is the least compared with GCVAE-I (with Maximum-Means Discrepancy-MMD distance between the density function of two continuous variables) and GCVAE-III (squared Mahalanobis distance between the density function of two continuous variables).

The Encoder–Decoder architecture for the GCVAE-II is shown in Fig. 4. 1-D Convolution with 64 layers firstly and 32 layers secondly along with max-pooling and batch normalization are applied in the encoder. Moreover, the same method is used with upsampling in the decoder. The Adam optimizer (Kingma & Ba, 2014) with a learning rate of $1 \times e^{-5}$ and a batch size of 64 is applied during GCVAE-II model training. After 200 iterations, we train the GCVAE-II model in 2 dimensions of the latent space.

Clustering the latent space of input and output by GMM

Since we have obtained the latent spaces with two dimensions for the FDS and FRCS in the above step. Clustering using GMM in the latent space of FDS and FRCS can be done separately. Here, all the 'keywords' of the original dataset are used ('keyword' can be 'short' or 'crack' etc.). After preprocessing, there are 12,032 observations, we only analyze 'Global success' = 'successful' to avoid the noise. After clustering by GMM into the latent space in FDS (x_n), 88 clusters are obtained as shown in Fig. 5. Using the same method, 26 clusters are obtained after clustering by GMM into the latent space in FRCS (y_n), as shown in Fig. 6. To evaluate clustering performance, six representative samples are selected in each group: one in the center, one at the outermost edge, and the remaining four samples are equidistant from the center to the edge. This is referred to as centroid analysis in the next section.

Centroid analysis

To simplify the analysis, we select only the data where the keyword is 'short'. For each cluster, six observations are selected: from the center to the edge, and the four samples between the center and the edge. The 6 observations from cluster 9 (99 observations in total) in the latent space of FDS (x_n) are presented in Table 3 (we list only the four features: **Reference**; **Subject**; **Context**; **Source of failure/request**):

These 6 observations are similar, the **Reference** have a similar format, and all mention **Customer Complaint** in the **Source of failure/request**. The same checks are performed for other clusters in FDS. We also check the 6 observations from cluster 4 in Table 4 (111 observations in total when the keyword='short') in the latent space of FRCS (y_n) (we list only the **Pt failure/Elt by sample**, **Macro failure mode by sample**, **Elementary failure mode**, **Tech cause/Defect by sample**, and **Analysis conclusion** here because we are only analyzing the **Global success = 'successful'**):

These 6 observations are similar, they all mention **BEoL_Metal** for the **Pt failure/Elt by sample**. And 5 observations (out of 6) talked about **Continuity failure** for the **Macro failure mode by sample**. Moreover, 4 observations (out of 6) mentioned **EOS** in the **Analysis conclusion**. The same verification is performed for the other clusters in FRCS. The clustering by GMM on the latent space obtained by GCAVE-II (MAH) is good enough to distinguish different groups in FDS (x_n) and in FRCS (y_n).

In the next step, we will apply AR to find the potential relationships between these 88 clusters in the latent space of FDS (x_n) and the 26 clusters in the latent space of FRCS (y_n).

Find the relationships between different clusters in the latent space of FDS and FRCS

As described in "Description of the method", AR aim to find the rules that determine the potential association between elements in a given transaction with a variety of elements. Each

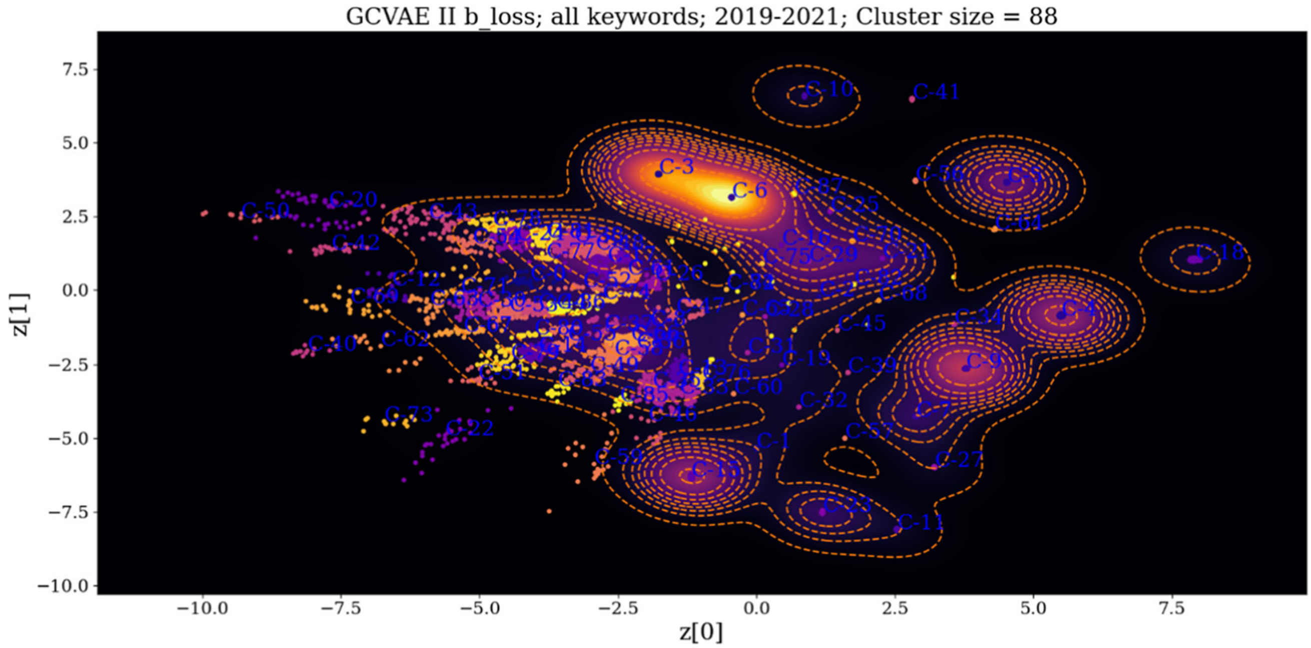


Fig. 5 Clustering in latent space with two dimensions of FDS (x_n), the optimal number of clusters is determined by BIC, 88 clusters are obtained in FDS for all keywords

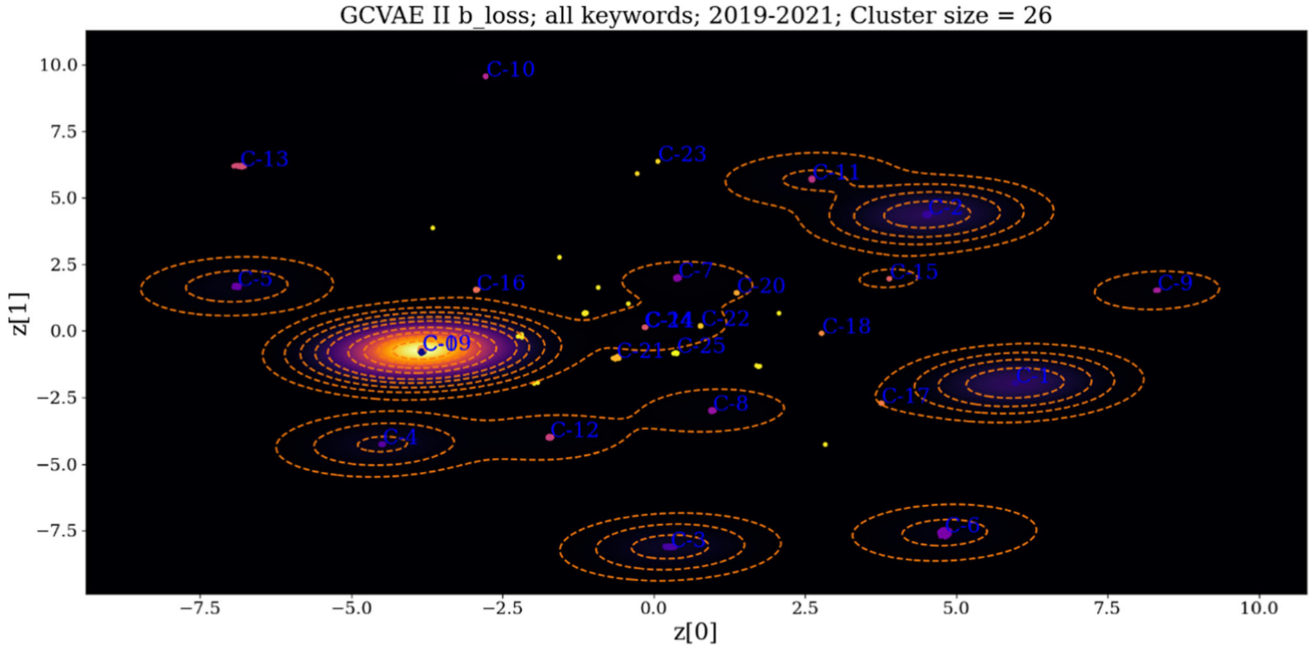


Fig. 6 Clustering in latent space with two dimensions of FRCS (y_n), the optimal number of clusters is determined by BIC, 26 clusters are obtained in FRCS for all keywords

observation has a label X_i in FDS (x_n) and simultaneously a label Y_j in FRCS (y_n). The goal is to propose a possible fault root cause conclusion (y_n) based on the fault description (x_n) by establishing a link between each label in FDS (X_i) and each label in FRCS (Y_j). The method consists of fixing the label in FDS (X_i) and then computing the three metrics: the $Support(X\_i \Rightarrow Y\_j)$, the $Conf(X\_i \Rightarrow Y\_j)$ and the $Lift(X\_i \Rightarrow Y\_j)$ for the 88 clusters in FRCS. First, only the clusters in FRCS Y_j whose $Lift(X\_i \Rightarrow Y\_j)$ greater than 1 are reserved. Then, the top 5 labels (Y_j) in FRCS are selected considering these three metrics.

For example, we set a cluster in FDS when $i = 6$ (X_6), which is the largest in FDS (1699 observations), then the three metrics are computed and ordered in Table 5 (we only list the

Table 3 Six observations from cluster 9 in the latent space of FDS (x_n) where the keyword is 'short'

Observations	Reference	Subject	Context	Source of failure/request
X1	F1938137570-PA2	Delphi_APTIV_UAR2-TR_GUAR2CEP_F1938137570_YL-19-098-PA2	The customer stated: Pin45 short to Pin48 resistance 52ohm	Customer Complaint
X2	F2004143069-PA1	ECC F2004143069	Failure customer description: short between A1 and A2. BTA10-600C-Trace code: GK827033	Customer Complaint
X3	F2002142423-PA2	Arrow_iRobot_STM32F303VET6_F446XXXY_F2002142423_QI-2019-24699	The customer stated: Abnormal: Pin73 GND short	Customer Complaint
X4	F1922133226-PA2	Kimball_SPC564A80L7CFAR_FA80CA_F1922133226_60062476	Pin125/Pin138 short to Pin43 (5 ohm)	Customer Complaint
X5	F1938137570-PA2	Delphi_APTIV_UAR2-TR_GUAR2CEP_F1938137570_YL-19-098-PA2	The customer stated: Pin45 short to Pin48 resistance 52ohm	Customer Complaint
X6	F1938137570-PA2	Delphi_APTIV_UAR2-TR_GUAR2CEP_F1938137570_YL-19-098-PA2	The customer stated: Pin45 short to Pin48 resistance 52ohm	Customer Complaint

Table 4 Six observations from cluster 4 in the latent space of FRCS (y_n) where **Global success** equals to 'successful'

Observations	Pt failure/Elt by sample	Macro failure mode by sample	Elementary failure mode	Tech cause/defect by sample	Analysis conclusion
Y1	BEoL_Metal	Continuity failure	Short	Melting in I/Os	EOS damage to power and output transistors
Y2	BEoL_Metal	Continuity failure	Resistive	Burnt	Physical-EOS damage
Y3	BEoL_Metal	Continuity failure	Resistive	Melting in I/Os	EOS
Y4	BEoL_Metal	Continuity failure	Short	Burnt	X-ray, excess solder, delamination at mold to die interface, continuity tests found shorted pins on each unit, packages decapped, visual inspection found burnt area in each die
Y5	BEoL_Metal	Continuity failure	Short	Melting in I/Os	EOS damage to power and output transistors
Y6	BEoL_Metal	Parametric failure	Short	Squash	Squash caused metal to short to nearby metal

labels whose metrics are not zero). According to Table 5, we select only the clusters Y_0, Y_22, Y_16 and Y_13 to propose the fault root cause conclusion of the analysis when a new observation is assigned to the cluster of X_6. We do not consider Y_20 because the $Lift(X\_6 \Rightarrow Y\_20) = 0.2951 < 1$ does not indicate any correlations between the X_6 cluster in FDS and the Y_20 cluster in FRCS. We can draw these four clusters in FRCS according to the three metrics as shown in Fig. 7.

To evaluate the performance of our method, we check each observation when the label in FDS is equal to 6 (X_6). There are 1699 observations. We check the ratio in which our proposal (4 labels in FRCS Y_0, Y_13, Y_16, Y_22) is correct for the label in FDS X_6 against the set of observations.

Table 5 Three metrics $Support(X\_6 \Rightarrow Y\_j)$, $Conf(X\_6 \Rightarrow Y\_j)$ and $Lift(X\_6 \Rightarrow Y\_j)$ for 5 labels whose values are not zero in FRCS when i = 6

X_6	$Support(X\_6 \Rightarrow Y\_j)$	$Conf(X\_6 \Rightarrow Y\_j)$	$Lift(X\_6 \Rightarrow Y\_j)$
Y_0	0.059	0.417	1.530
Y_13	0.0004	0.0029	2.3606
Y_16	0.0006	0.0041	1.9066
Y_20	8.3e−5	0.0006	0.2951
Y_22	0.0813	0.5756	1.6266

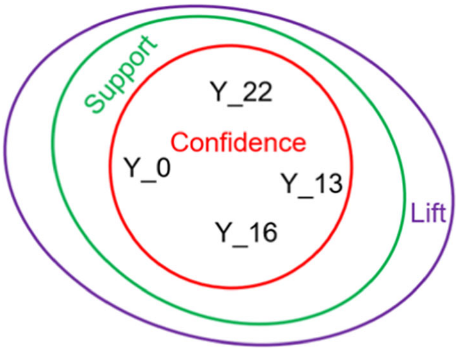


Fig. 7 Four clusters suggested in FRCS when FDS label is set as X_6

Table 6 The correct ratio if the label of FRCS is the same as what we predicted for the same ID observation if X_6

With the same ID	Y_0	Y_13	Y_16	Y_22
If X_6	41.7%	0.29%	0.4%	57.5%

Table 7 Three metrics $Support(X\_3 \Rightarrow Y\_j)$, $Conf(X\_3 \Rightarrow Y\_j)$ and $Lift(X\_3 \Rightarrow Y\_j)$ for 6 labels whose values are not zero in FRCS when i = 3

X_3	$Support(X\_3 \Rightarrow Y\_j)$	$Conf(X\_3 \Rightarrow Y\_j)$	$Lift(X\_3 \Rightarrow Y\_j)$
Y_0	0.044	0.417	1.532
Y_1	0.0005	0.0047	0.0536
Y_6	8.3e-5	0.0008	0.2369
Y_8	8.3e-5	0.0008	0.4119
Y_19	8.3e-5	0.0008	0.7895
Y_22	0.0607	0.5756	1.6265

Table 6 shows if the label of FDS is X_6, the true label of the same observation in FRCS is within the predicted labels. Taking the sum of all these ratios, we get 99.99% as the correct ratio if the true label of FRCS is within our 4 clusters computed by AR.

Then we fix the label X_3 which is the second most frequent (1270 observations) in FDS, and the correct ratio is also calculated if the true label of FRCS is within our 2 proposed clusters (Y_0 and Y_22 in this case), which is 99.29%. We can see the table of individual metrics computed for each label in FRCS when $i = 3$ (X_3) in Table 7.

Table 8 Three metrics $Support(X\_87 \Rightarrow Y\_j)$, $Conf(X\_87 \Rightarrow Y\_j)$ and $Lift(X\_87 \Rightarrow Y\_j)$ for 2 labels whose values are not zero in FRCS when i = 87

X_87	$Support(X\_87 \Rightarrow Y\_j)$	$Conf(X\_87 \Rightarrow Y\_j)$	$Lift(X\_87 \Rightarrow Y\_j)$
Y_0	0.0003	0.5	1.835
Y_22	0.0003	0.5	1.4129

Table 9 Three metrics $Support(X\_64 \Rightarrow Y\_j)$, $Conf(X\_64 \Rightarrow Y\_j)$ and $Lift(X\_64 \Rightarrow Y\_j)$ for 2 labels whose values are not zero in output space when i = 64

X_64	$Support(X\_64 \Rightarrow Y\_j)$	$Conf(X\_64 \Rightarrow Y\_j)$	$Lift(X\_64 \Rightarrow Y\_j)$
Y_0	0.0003	0.4444	1.6313
Y_22	0.0004	0.5556	1.5699

We also fix the label X_87 that has the smallest number (8 observations) in FDS and compute the correct ratio if the true label of FRCS is within the 2 clusters we proposed (Y_0 and Y_22 in this case), which is 100%. We can see the table of individual metrics computed for each label in FRCS when $i = 87$ (X_87) in Table 8.

We also fix the label X_64 that has the second lowest number (9 observations) in FDS, and compute the correct ratio if the true label of FRCS is within the 2 clusters we proposed (Y_0 and Y_22 in this case) which is 100%. We can see the table of individual metrics computed for each label in FRCS when $i = 64$ (X_64) in Table 9. According to the ratio of these 4 clusters in FDS (X_6, X_3, X_87, X_64), we get a correct ratio of more than 99%, which is satisfactory.

In the end, if the label of FRCS is within the array we predicted for the same ID of observation for each label in FDS, the correct ratio is computed according to the business rules proposed in "Description of the method". The correct ratio and the number of observations for each cluster in FDS are shown in Table 10 (from *X_0* to *X_29*), Table 11 (from *X_30* to *X_59*) and Table 12 (from *X_60* to *X_87*). The mean of the correct ratio for all the labels in FDS (x_n) can then be calculated as 97.8%, which is satisfactory to help industry experts predict the Fault Analysis (FA) conclusion. For example, the

Table 10 The correct ratio when the label of FRCS is inside the field we have proposed for the same ID of observation for each label in FDS: from X_0 to X_29

Labels in input space	Correct ratio (%)	Number of observations
X_0	100	36
X_1	100	134
X_2	93.2	118
X_3	99.3	1270
X_4	93.9	738
X_5	96.8	536
X_6	99.9	1699
X_7	94.8	270
X_8	97.4	115
X_9	92.2	892
X_10	94.8	136
X_11	100	102
X_12	96.0	25
X_13	93.1	58
X_14	95.7	93
X_15	100	697
X_16	100	271
X_17	97.1	139
X_18	97.4	154
X_19	98.3	60
X_20	100	41
X_21	100	350
X_22	96.7	30
X_23	94.3	282
X_24	98.8	81
X_25	95.1	164
X_26	98.9	94
X_27	96.3	108
X_28	95.8	96
X_29	98.9	281

Table 11 The correct ratio when the label of FRCS is inside the field we have proposed for the same ID of observation for each label in FDS: from X_30 to X_59

Labels in input space	Correct ratio (%)	Number of observations
X_30	97.3	73
X_31	92.9	156
X_32	95.3	107
X_33	100	87
X_34	100	87
X_35	96.7	61
X_36	100	42
X_37	97.4	39
X_38	95.4	151
X_39	97.5	80
X_40	100	12
X_41	100	19
X_42	100	22

Table 11 continued

Labels in input space	Correct ratio (%)	Number of observations
X_43	98.5	67
X_44	98.7	76
X_45	100	45
X_46	100	15
X_47	100	62
X_48	95.6	184
X_49	100	49
X_50	100	11
X_51	100	17
X_52	100	19
X_53	98.8	87
X_54	94.5	55
X_55	92.6	54
X_56	100	11
X_57	90	20
X_58	97.3	111
X_59	100	25

Table 12 The correct ratio when the label of FRCS is inside the field we have proposed for the same ID of observation for each label in FDS: from X_60 to X_87

Labels in input space	Correct ratio (%)	Number of observations
X_60	100	31
X_61	97.0	135
X_62	100	16
X_63	100	28
X_64	100	9
X_65	100	70
X_66	92.6	163
X_67	100	30
X_68	100	13
X_69	100	23
X_70	100	23
X_71	100	23
X_72	100	14
X_73	100	13
X_74	96.9	66
X_75	100	28
X_76	100	39
X_77	100	21
X_78	100	46
X_79	92.9	57
X_80	100	18
X_81	91.3	81
X_82	100	16
X_83	100	20
X_84	97.2	36
X_85	90.5	21
X_86	97.1	70
X_87	100	8

Context is about ‘PinXX short to PinXX’ while the **Source of failure/request** is always ‘Customer Complaint’ when the label in the latent space of FDS (x_n) is 9 (cluster *X_9*) as mentioned in Table 3. According to our intelligent Fault Analysis system, we propose that the fault root cause could be Electrical Over-Stress (EOS) damage.

Conclusions and perspectives

In this paper, a complete AI-based knowledge extraction pipeline for Fault Analysis in semiconductor industry is proposed following a data-driven approach. Using advanced NLP and Machine Learning (ML) techniques, text data from expert analysis reports are processed and the most relevant information is extracted into latent spaces and clustered by Gaussian Mixture Model (GMM). Then, Association Rules (AR) are established to find the relationship between the clustered latent space of fault description and the fault root cause. The proposed AR also contribute to better interpretability of AI. The complete architecture can automatically predict the fault root cause based on the fault description, which can significantly improve the efficiency of FA while reducing the potential cost of inspection exploration actions. The overall algorithm is evaluated with real industry data over three years and the mean correctness of the predicted label is 97.8%.

The main limitation of the proposed method is that the model can only predict the fault root causes existing in the current training database. That means if new failures or failures related to multiple root causes occur, the model would still predict one previously existing cause label. This limitation can be handled on the one hand by training a reinforcement learning model, which is able to learn new fault descriptions and the root cause in a continuous way. On the other hand, training a Natural Language Generative model to generate root cause description instead of a classification model can potentially cover the failure root causes related to multiple classes. These two approaches are the future research work which can improve the robustness of the proposed algorithm. Moreover, because the proposed algorithm depends strongly on the quality of the clustering y GMM, for future work, it would be also interesting to combine some (not all) features of the Analysis Path Space to enhance the clustering performances.

Acknowledgements This research is financially supported by EURIPIDES2 under the PENTA framework of the European project Fault Analysis (FA) 4.0 EUREKA Cluster. The authors would like to acknowledge the expertise provided by ST Microelectronics, in particular, the discussions and data acquisition support provided by Pascal Gounet and Jerome Adrian.

Author contributions All authors contributed to the study conception and design. Data collection were performed by Kenneth Ezukwoke. Data analysis and the algorithm design were performed by Zhiqiang Wang. The first draft of the manuscript was written by Zhiqiang Wang and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding This research is financially supported by EURIPIDES2 under the PENTA framework of the European project Fault Analysis (FA) 4.0 EUREKA Cluster.

Data availability The datasets analysed during the current study are not publicly available due confidential company data by STMicroelectronics but are available from STMicroelectronics (pascal.gounet@st.com) on reasonable request.

Declarations

Conflict of interest The authors declare that they have no conflict of interest and have no competing interests.

Appendix A: List of abbreviation

Abbreviation	Definition
ADC	Automatic defect classification
AI	Artificial intelligence
ANN	Artificial neural network
APS	Analysis paths space
AR	Association rules
BIC	Bayesian information criterion
CNNs	Convolutional neural networks
DLHRR	Deep-learning for high-resolution reconstruction
EOS	Electrical over-stress
ESD	ElectroStatic discharge
FA	Fault analysis
FATs	Fault analysis triplets
FDS	Fault description space
FM	Failure mechanism
FP-Growth	Frequent pattern growth
FRACAS	Fault reporting, analysis, and corrective action system
FRCS	Fault root cause space
GCVAE	Generalized-controllable variational AutoEncoder
GMM	Gaussian mixture model
LM	Language models
LSI	Large-scale integration
MMD	Maximum-means discrepancy
MAH	MaHalanobis distance
NLP	Nature language processing
PCA	Principal component analysis
RCA	Root cause analysis
SVC	Support vector classifier
VAE	Variational AutoEncoder
XRM	X-ray microscopy
3DCNN	Three-dimensional convolutional neural network

References

Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on management of data* (pp. 207–216). Association for Computing Machinery, New York, NY, United States.

Antomarioni, S., Ciarapica, F. E., & Bevilacqua, M. (2022). Association rules and social network analysis for supporting failure mode effects and criticality analysis: Framework development and insights from an onshore platform. *Safety Science, 150*, 105711.

Antonello, F., Baraldi, P., Shokry, A., Zio, E., Gentile, U., & Serio, L. (2021). Association rules extraction for the identification of functional dependencies in complex technical infrastructures. *Reliability Engineering & System Safety, 209*, 107305.

Bajenescu, T. I., & Bazu, M. I. (2012). *Reliability of electronic components: A practical guide to electronic systems manufacturing*. Springer.

Dimitriou, N., Leontaris, L., Vafeiadis, T., Ioannidis, D., Wotherspoon, T., Tinker, G., & Tzovaras, D. (2019). Fault diagnosis in microelectronics attachment via deep learning analysis of 3-D laser scans. *IEEE Transactions on Industrial Electronics, 67*(7), 5748–5757.

Ezukwoke, K., Hoayek, A., Batton-Hubert, M., & Boucher, X. (2022). GCVAE: Generalized-controllable variational autoencoder. https://doi.org/10.48550/ARXIV.2206.04225.

Ezukwoke, K., Toubakh, H., Hoayek, A., Batton-Hubert, M., Boucher, X., & Gounet, P. (2021). Intelligent fault analysis decision flow in semiconductor industry 4.0 using natural language processing with deep clustering. In *2021 IEEE 17th international conference on automation science and engineering (CASE)* (pp. 429–436). IEEE.

Grabner, C., Safont-Andreu, A., Burmer, C., & Schekotihin, K. (2022). A BERT-based report classification for semiconductor failure analysis. In *ISTFA 2022* (pp. 28–35). ASM International.

Gu, A., Terada, M., Stegmann, H., Rodgers, T., Fu, C., & Yang, Y. (2022). From system to package to interconnect: An artificial intelligence powered 3d x-ray imaging solution for semiconductor package structural analysis and correlative microscopic failure analysis. In: *2022 IEEE international symposium on the physical and failure analysis of integrated circuits (IPFA)* (pp. 1–5). IEEE.

Hahsler, M., Grün, B., & Hornik, K. (2005). arules—A computational environment for mining association rules and frequent item sets. *Journal of Statistical Software, 14*, 1–25.

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science, 313*(5786), 504–507.

Hollnagel, E. (2018). *Safety-I and safety-II: The past and future of safety management* (pp. 61–90). CRC Press.

Hsu, C.-Y., & Liu, W.-C. (2021). Multiple time-series convolutional neural network for fault detection and diagnosis and empirical study in semiconductor manufacturing. *Journal of Intelligent Manufacturing, 32*, 823–836.

Ji, C.-R., & Deng, Z.-H. (2007). Mining frequent ordered patterns without candidate generation. In *Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)* (Vol. 1, pp. 402–406). IEEE.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint. https://doi.org/10.48550/arXiv.1412.6980.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. arXiv preprint. https://doi.org/10.48550/arXiv.1312.6114.

Mangalathu, S., & Jeon, J.-S. (2019). Machine learning-based failure mode recognition of circular reinforced concrete bridge columns: Comparative study. *Journal of Structural Engineering, 145*(10), 04019104.

Martin, P. L. (1999). *Electronic failure analysis handbook: Techniques and applications for electronic and electrical packages, components, and assemblies*. McGraw-Hill Education.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint. https://doi.org/10.48550/arXiv.1301.3781.

Nagamura, Y., Arima, K., Arai, M., & Fukumoto, S. (2021). Layout feature extraction using CNN classification in root cause analysis of LSI defects. *IEEE Transactions on Semiconductor Manufacturing, 34*(2), 153–160.

Phua, C., & Theng, L. B. (2020). Semiconductor wafer surface: Automatic defect classification with deep CNN. In *2020 IEEE region 10 conference (TENCON)* (pp. 714–719). IEEE.

Platter, F., Safont-Andreu, A., Burmer, C., & Schekotihin, K. (2021). Report classification for semiconductor failure analysis. In *ISTFA 2021* (pp. 1–5). ASM International.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. arXiv preprint. https://doi.org/10.48550/arXiv.1908.10084 https://doi.org/10.48550/arXiv.1908.10084

Reynolds, D. A., et al. (2009). Gaussian mixture models. Encyclopedia of. *Biometrics, 741*, 659–663.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics, 6*(2), 461–464. https://doi.org/10.1214/aos/1176344136

Trappey, A., Trappey, C. V., & Hsieh, A. (2021). An intelligent patent recommender adopting machine learning approach for natural language processing: A case study for smart machinery technology mining. *Technological Forecasting and Social Change, 164*, 120511.

Wang, Z., Ezukwoke, K., Hoayek, A., Batton-Hubert, M., & Boucher, X. (2022). NLP based on GCVAE for intelligent fault analysis in semiconductor industry. In *2022 IEEE 27th international conference on emerging technologies and factory automation (ETFA)* (pp. 1–8). IEEE.

Wang, Z., Ritou, M., Da Cunha, C., & Furet, B. (2020). Contextual classification for smart machining based on unsupervised machine learning by Gaussian mixture model. *International Journal of Computer Integrated Manufacturing, 33*(10–11), 1042–1054.

Watanabe, A., Hirose, N., Kim, H., & Omura, I. (2019). Convolutional neural network (CNNs) based image diagnosis for failure analysis of power devices. *Microelectronics Reliability, 100*, 113399.

Xiao, S., Hu, Y., Han, J., Zhou, R., & Wen, J. (2016). Bayesian networks-based association rules and knowledge reuse in maintenance decision-making of industrial product-service systems. *Procedia Cirp, 47*, 198–203.

Xie, Y., Li, S., Wu, C., Lai, Z., & Su, M. (2022). A novel hypergraph convolution network for wafer defect patterns identification based on an unbalanced dataset. *Journal of Intelligent Manufacturing*, 1–14. https://doi.org/10.1007/s10845-022-02067-z.

Xu, Z., & Saleh, J. H. (2021). Machine learning for reliability engineering and safety applications: Review of current status and future opportunities. *Reliability Engineering & System Safety, 211*, 107530.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Terms and Conditions