CoFa: Contrasting Matching with Probabilistic Fault Modeling for Optimizing Task Assignment in IC Testing

***Abstract*—As integrated circuit (IC) processes become more complex and the challenge to meet high market demands grows, enhancing yield has become a critical factor for the semiconductor industry. Identifying potential IC defect types and allocating the corresponding engineers to address them can mitigate losses, thereby expediting time-to-market. So far existing approaches often ignore the time consumption required to handle a task and competencies of individual engineers. Motivated by this, we propose CoFa (Contrasting Matching with Probabilistic Fault Modeling), an innovative approach that models the probabilities of defect types in integrated circuit modules through a graph structure and utilizes a contrasting matching method to identify the most suitable or proficient engineers. This design not only optimizes task assignment in IC testing but also enhances human resource management by ensuring tasks are handled by the most competent engineers, balancing workloads, and promoting diverse experience in addressing different defect types. Through experiments on a real-world dataset, we demonstrate that CoFa achieves the highest number of successful cases in assigning appropriate engineers to handle various defect types of integrated circuit modules compared to baselines. Additionally, it balances each engineer's workload and considers the variety of defect types they handle.**

I. Introduction

With the global growth in the production and demand for integrated circuits (ICs), covering multiple specialized fields from design to production, minimizing the time from production to shipment while ensuring quality at each stage has become a crucial issue in the semiconductor industry. Each stage, from design and process to sample delivery for mass production, requires passing through its own testing phases before proceeding to the next stage. Therefore, the time and quality of testing at each pre-production stage are interlinked. Enhancing testing speed while maintaining high quality has become a significant challenge.

Generally, when manufacturers produce samples and deliver them to various customers, they receive numerous return material authorizations (RMAs) [1] . RMAs are closely linked to the development of the next phase of ICs because when customers encounter problems and return RMAs to the quality assurance (QA) team, the QA and test verification departments clarify the issues and address them, impacting the development of the next phase. This close relationship with IC design and production speed becomes a key competitive factor.

The test flow is shown in Fig. 1. In the testing department, RMA first undergoes golden version testing through the equipment, with each RMA assigned as a different task handled by engineers. After testing with automatic test equipment (ATE)

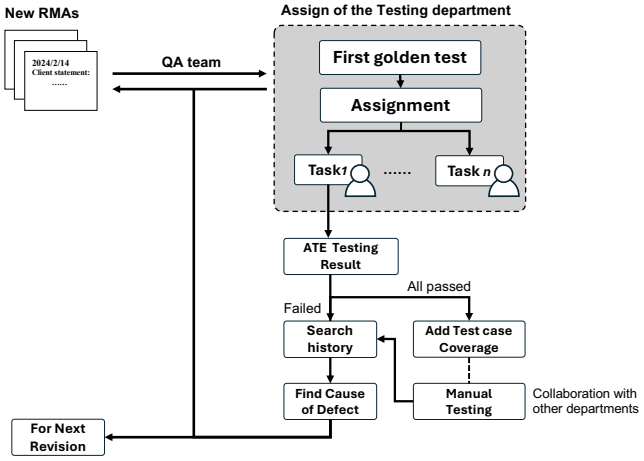


Fig. 1: RMA process example diagram.

[2], a test report as log file is generated, containing results of many test cases. Engineers identify chip defects based on the failed test cases. The fastest way to address issues is to search past data based on previous experience and attempt to determine the root cause of the problem. Once identified, the engineer adjusts and reports the final results to the QA team. Since the process heavily relies on the engineer's past experience with similar issues, it is crucial to assign tasks (RMA) to the appropriate engineers, as it directly impacts the speed of problem resolution.

Currently, task assignment problem is predominantly managed manually by supervisors, leading to several challenges: Firstly, task type definition and engineer experience are critical issues. Initially, only ATE test results are available, making it difficult to quickly classify a task type and identify engineers with relevant expertise. Secondly, individual workload management is essential, as excessive assignments can hinder an engineer's efficiency and productivity. Finally, there is the issue of learning opportunities for engineers. Overspecialization in certain types of tests can limit the flexibility of task allocation and reduce opportunities for engineers to gain diverse experience.

Recent works have increasingly focused on automatic fault localization in IC testing [3][4], primarily aiming to identify potential fault locations or types during the manufacturing process or improving detection yield at the design stage to reduce costs and time [5][6]. However, these approaches encounter several challenges: (i) certain defect types in ICs may

TABLE I: Overview of the data structure of an RMA and its ATE logs.

File Category	Variable	Description	Example
RMA information	$rma.id$	The individual numeric identifier for the received RMA from customers.	432119Q
	$rma.start$	The date (year, month, and day) on which an RMA is received.	2020-11-25
	$rma.close$	The date (year, month, and day) on which an RMA was completed.	2020-12-20
	$rma.eng$	The name of the engineer responsible for handling the RMA.	Teila
	$defect.type$	The primary defect type causing the IC failure in an RMA.	#EIPD
ATE logs of an RMA	$test.id$	The individual numeric identifier for the test item.	1158
	$test.name$	The name for the test item.	Ripple_L04_-VCC
	$limit.min$	The minimum value of the normal test range for this test item.	0.000 dB
	$measure$	The measured value of the test item.	0.225 dB
	$test.result$	The test results of the test item.	passed

exhibit similar failure characteristics, necessitating require the expertise of engineers who have experience with similar failure information to perform accurate fault diagnosis. (ii) After detecting an IC issue, it is essential to report the fault to quality assurance teams, communicate with customers, or send the IC back to the manufacturer for repairs. Detailed fault reports help customers understand the issue and assist manufacturers in efficiently repairing the ICs. Therefore, assigning tasks to engineers with relevant domain knowledge is crucial for providing comprehensive fault descriptions and ensuring smooth operations. In contrast to prior studies, our methodology emphasizes the optimization of processes to minimize temporal expenditures during the manual testing phases, which is a domain that remains unexplored within contemporary IC testing research.

To address the mentioned issues and ensure a seamless IC manufacturing process and RMA handling, we propose a novel framework **CoFa** (Contrasting Matching with Probabilistic Fault Modeling). The objective is to determine the optimal RMA task assignment to maximize the task handling success rates while considering the engineers' workloads and the diversity of knowledge they acquire. Specifically, *CoFa* incorporates two key strategies. First, to learn the probability of an RMA task belonging to different defect types, a defect-aware graph is constructed and an extended embedding model is designed to better capture the failure characteristics of an RMA task. Second, to determine the most appropriate engineer to handle the RMA tasks, a dynamic selection process is utilized to determine candidates and a contrasting-based method is deployed to select the most suitable engineer. Our method effectively optimizes the task assignment by designing a novel probabilistic fault modeling method.

In *CoFa*, we utilize ATE-obtained test results (ATE logs) to conduct a preliminary root cause analysis. By analyzing the high similarity between failed test items, we use graph embedding techniques to transform issue types into similarity representations. Based on these representations, we allocate tasks according to engineers' past experience with similar task types, while also considering each engineer's parallel processing capabilities to optimize the overall task allocation time. We aim to analyze engineers' proficiency in handling different chip defect issues to expedite the task processing speed. Our key contributions are summarized as follows:

- We introduce the *CoFa* framework, which utilizes a novel graph-based modeling method to capture the failure characteristics and the probabilistic defect type of each RMA task.
- To determine the optimal assignment, a contrasting-based dynamic selection process is designed to identify engineers who are most likely familiar with similar failure characteristics.
- We perform experiments on a real-world dataset, demonstrating the robustness and practicality of *CoFa* compared to baseline methods.

II. Problem Settings

A. Dataset

For the purpose of this study, we utilize real data from a semiconductor company. The data includes received RMAs for IC failures from customers and the ATE logs for each RMA from 2020 to 2023, handled by four engineers. Table I lists the key data fields, where some fields such as $rma.id$, $rma.eng$, and $test.id$ have been anonymized. RMAs are categorized into $defect.type$ by the engineers responsible for each RMA after analyzing the results of the ATE logs. The contents of the ATE logs include multiple test items and the normal test range ($limit.min$) for each test item. If the measured value ($measure$) obtained from the ATE exceeds the normal test range, it indicates that the RMA has failed this test item.

B. Analysis on Failure Characteristics

Based on the information shown in Table I, we conduct a preliminary analysis to validate the notion that different defect types may exhibit similar failure characteristics, as mentioned in Sec. I. Figure 2 shows the fail rate of each test name for different defect types of RMAs across all ATE logs. We can observe that although different defect types have varying fail rates across test names, some defect types exhibit similar failure characteristics while others differ significantly.

1) Similar Characteristics: Defect types "*#MIM_capacitor*" and "*#Die_damage*" (highlighted by the red dotted line in Figure 2) have a nearly zero fail rate between test name indices 34 to 43, and their fail rate distributions are also quite similar for other test name

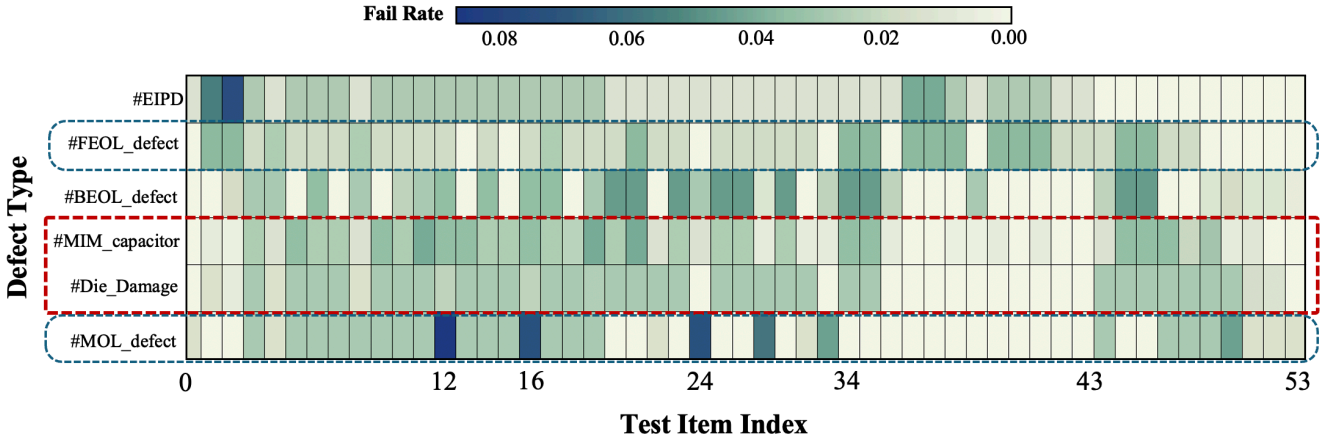


Fig. 2: Fail rates of each defect type of IC modules on different test names in ATE logs.

indices. The type "*#Die_damage*" refers to the physical damage sustained by the chip (die) itself during the wafer manufacturing or packaging process of ICs. On the other hand, "*#MIM_capacitor*" refers to issues with metal-insulator-metal capacitors, such as capacitance deviation from specifications or defects in the dielectric layer. Both defect types may originate from problems in the manufacturing process, such as improper handling during manufacturing or packaging, or material and process control issues during manufacturing. As a result, they may fail similar test items and have similar failure characteristics.

2) Dissimilar Characteristics: However, some defect types exhibit significantly different failure characteristics. The defect types "*#FEOL_defect*" and "*#MOL_defect*" (highlighted by the blue dotted line in Figure 2) show substantial differences in their fail rate distributions. The type "*#FEOL_defect*" has higher fail rates in test item indices 34 to 43 compared to "*#MOL_defect*", whereas the type "*#MOL_defect*" shows much higher fail rates than "*#FEOL_defect*" at indices 12, 16, and 24. This discrepancy arises because the type "*#FEOL_defect*" refers to defects occurring in the Front-End-Of-Line (FEOL) stage of wafer manufacturing, primarily affecting the electrical performance of transistors. In contrast, "*#MOL_defect*" pertains to defects in the Middle-Of-Line (MOL) stage, mainly impacting signal transmission and interconnect resistance. Since these defect types occur at different manufacturing stages, their failure characteristics are not similar.

This finding indicates that different defect types have varying degrees of correlation in their failure characteristics, which may only appear in specific test items. This highlights the complexity of correlations between defect types. Consequently, these observations demonstrate the necessity of capturing correlations between failures based on defect types and failed test items to ensure each RMA task is assigned to the most experienced and appropriate engineer.

C. Optimized Task Assignment Problem

In our paper, we treat each RMA initiated by customers as a task that needs to be handled. Our objective is to assign each RMA to the appropriate engineer, maximizing the success rate of resolving the assigned RMA before the customer's expected time. Additionally, we aim to balance the workload of each engineer and ensure a diverse range of defect types in the RMAs (tasks) they handle. This approach allows engineers to gain experience with different defect types within their capacity. We formulate our RMA task assignment as follows. **Problem Statement:** Given a set of RMA tasks $\mathcal{V}_{task}$ and a set of engineers E, each RMA task $t \in \mathcal{V}_{task}$ comes with its ATE log testing results and defect type, and each engineer $e \in E$ comes with information on previously handled RMAs. During the testing phases, we sequentially receive new RMA tasks. Our aim is to assign these tasks to the appropriate engineers in a way that maximizes the success rate, while also considering the engineers' current workloads and the diversity of the defect types they have handled.

In the remainder of this paper, we will describe the proposed *CoFa*, which captures the correlations between the failure characteristics of RMA tasks. We will further utilize a contrast-based method to determine the most appropriate engineer to handle a newly received RMA task.

III. Proposed Method: CoFa

The architecture of *CoFa* consists of three main components. Firstly, we formulate a defect-aware graph to preserve the information of ATE logs of each RMA task. Based on the historical records of RMA tasks and the defect-aware graph, the defect-aware representations of RMA tasks and engineers are formulated. Using these representations, a dynamic selection process is employed to filter the similarity between engineers and RMA tasks based on their compatibility with each defect type. Finally, according to the selected engineer candidates, the matching score of each candidate engineer to a newly-received RMA task is estimated to determine the most appropriate task assignment plan.

A. Defect-Aware Representation

1) Engineer Representation: To evaluate the experience and ability of each engineer in handling different kinds of defect types of ICs, we first formulate the normalized representation $\bar{\mathbf{e}}_i$ for each engineer $e_i \in E$ based on their previously handled RMAs information. Given the defect type set $\mathcal{V}_{type}$, let the average processing time and the number of times defect type $p \in \mathcal{V}_{type}$ has been handled by engineer $e_i \in E$ be $\mathcal{P}_{i,p}$ and $\mathcal{N}_{i,p}$, respectively. The ability of an engineer e_i to handle defect type p is defined as follows:

$$e_{i,p} = \text{avg}(1 - \frac{\mathcal{P}_{i,p}}{\sum_{e_j \in E} \mathcal{P}_{j,p}/|E|}\ ,\ \frac{\mathcal{N}_{i,p}}{\sum_{e_j \in E} \mathcal{N}_{j,p}/|E|}). \quad (1)$$

Based on the defined ability $e_{i,p}$ of engineer e_i, the normalized defect-aware representation $\bar{\mathbf{e}}_i \in \mathbb{R}^{|\mathcal{V}_{type}|}$ of engineer e_i is formulated as follows:

check formula

$$\bar{\mathbf{e}}_i = \left[\frac{e_{i,p}}{|\mathbf{e}_i|}\right]_{p \in \mathcal{V}_{type}} ; \mathbf{e}_i = [e_{i,p}]_{p \in \mathcal{V}_{type}}. \quad (2)$$

2) RMA Task Representation: To preserve the ATE logs information and model the failure characteristics of RMA tasks and defect types, we design a defect-aware graph based on the ATE logs.

Definition 1 (Defect-Aware Graph): Given the RMA task set $\mathcal{V}_{task}$, defect type set $\mathcal{V}_{type}$, and the test item set $\mathcal{V}_{test}$, the relationships between these entities are defined as follows. From the ATE logs of RMA tasks, if test item $m \in \mathcal{V}_{test}$ fails in the ATE logs of RMA task $t \in \mathcal{V}_{task}$, then $\varepsilon_{m \leftrightarrow t} = 1$ holds. This relationship forms the edge set $\mathcal{E}_{test \leftrightarrow task} = \{\varepsilon_{m \leftrightarrow t} | m \in \mathcal{V}_{test}, t \in\in \mathcal{V}_{task}\}$. If RMA task $t \in \mathcal{V}_{task}$ belongs to defect type $p \in \mathcal{V}_{type}$, then $\varepsilon_{t \leftrightarrow p} = 1$ holds. This relationship forms the edge set $\mathcal{E}_{task \leftrightarrow type} = \{\varepsilon_{t \leftrightarrow p} | t \in \mathcal{V}_{task}, p \in\in \mathcal{V}_{type}\}$. Therefore, the defect-aware graph $\mathcal{G}$ is defined as follows:

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X}) \begin{cases} \mathcal{V} = \mathcal{V}_{test} \cup \mathcal{V}_{task} \cup \mathcal{V}_{type} \ \ (nodes) \\ \mathcal{E} = \mathcal{E}_{test \leftrightarrow task} \cup \mathcal{E}_{task \leftrightarrow type} \ \ (edges) \\ \mathcal{X} = \{x_v | v \in \mathcal{V}\} \ \ (attributes) \end{cases}, \quad (3)$$

where $\mathcal{X}$ is an attribute set, and each $x_v \in \mathcal{X}$ represents a node attribute of "*test item*", "*RMA task*" or "*defect type*".

To represent the characteristics of each node in the defect-aware graph $\mathcal{G}$, we utilize the classic model DeepWalk [7] and extend the skip-gram architecture [8][9] by considering the second-order proximity [10] of each node in defect-aware graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X})$. Therefore, the optimization function is redesigned as follows:

$$\max_{\psi} \sum_{u \in \mathcal{V}} \Big[-\log Z_u + \sum_{v \in N(u)} \Big(1 + \omega_{v,u}\Big) \cdot \Big(\psi(v) \cdot \psi(u)\Big) \Big],$$

$$\omega_{v,u} = \begin{cases} J\big(N(v) \cap \mathcal{V}_{test}, N(u) \cap \mathcal{V}_{test}\big), \text{ if } v \neq u \text{ and } v, u \in \mathcal{V}_{task} \\ 0 \qquad\qquad\qquad\qquad\qquad\quad , \text{otherwise} \end{cases} \quad (4)$$

where $Z_u = \sum_{u \in \mathcal{V}} \exp(\psi(v) \cdot \psi(u))$ can be approximated by negative sampling [11]. The function $J(\cdot, \cdot)$ represents the Jaccard similarity of two sets. $\psi(v) \in \mathbb{R}^d$ and $\psi(u) \in \mathbb{R}^d$ are embedding vectors of nodes v and u, respectively.

Based on Eq. 4, the representation of a newly-received task t_k ($\notin \mathcal{V}_{task}$) can be formulated by considering the failed test items in its ATE logs. Let $\mathcal{F}_k \subset \mathcal{V}_{test}$ includes the failed test items in the ATE logs of new task t_k. The probability of RMA task t_k belongs to defect type $p \in \mathcal{V}_{type}$ is estimated as follows:

$$s_{k,p} = sim\Big(\frac{1}{|\mathcal{F}_k|} \sum_{m \in \mathcal{F}_k} \psi(m),\ \psi(p)\Big), \quad (5)$$

where the function $sim(\cdot, \cdot)$ represents the cosine similarity. Based on Eq. 5, the normalized defect-aware representation $\bar{\mathbf{t}}_k \in \mathbb{R}^{|\mathcal{V}_{type}|}$ of newly-received RMA task t_k is formulated as follows:

$$\bar{\mathbf{t}}_k = \left[\frac{s_{k,p}}{|\mathbf{t}_k|}\right]_{p \in \mathcal{V}_{type}} ; \mathbf{t}_k = [s_{k,p}]_{p \in \mathcal{V}_{type}}. \quad (6)$$

B. Dynamic Candidate Selection

According to Eq. 2 and Eq. 6, the obtained defect-aware representations $\bar{\mathbf{e}}_i$ and $\bar{\mathbf{t}}_k$ of each engineer $e_i \in E$ and the newly received task t_k are used to select the candidate engineers who may have experience handling tasks with similar failure characteristics to t_k. Therefore the candidate engineer set E_{cand} can be formulated as follows:

$$E_{cand} = \{e_i \in E \mid d(e_i, t_k) \geq \vartheta\}, \text{where} \quad (7)$$

$$d(e_i, t_k) = \text{JSD}(\bar{\mathbf{e}}_i || \bar{\mathbf{t}}_k), \quad (8)$$

where $\text{JSD}(\cdot, \cdot)$ is the Jensen-Shannon divergence, and ε is a fixed threshold.

C. Contrasting-Based Assignment

Given the candidate engineer set E_{cand} obtained from Sec. III-B, we aim to assign the most appropriate engineer to handle the new task t_k. For each engineer $e_i \in E_{cand}$, the matching score $\mu_{i,k}$ for handling t_k can be estimated by contrasting their representation values for each defect type. The matching score $\mu_{i,k}$ is designed as follows:

$$\mu_{i,k} = \sum_{p \in \mathcal{V}_{type}} \text{softmax}\big(\mathcal{S}(e_i, t_k, p)\big), e_i \in E_{cand}, \quad (9)$$

$$\mathcal{S}(e_i, t_k, p) = \begin{cases} \log \dfrac{\bar{\mathbf{e}}_i(p)}{\bar{\mathbf{t}}_k(p)} & , \text{if } N_i(p) \geq \gamma \\ -\infty & , \text{otherwise} \end{cases}, \quad (10)$$

where $\bar{\mathbf{e}}_i(p) = e_{i,p}$ represents the ability of engineer e_i for defect type p (Eq. 1). $\bar{\mathbf{t}}_k(p) = t_{k,p}$ indicates the probability that task t_k belongs to defect type p (Eq. 5). $N_i(p)$ denotes the number of tasks of defect type p handled by engineer e_i. The parameter γ is a fixed value.

Let $M = \{(e_i, \mu_{i,k}) | e_i \in E_{cand}\}$ be a set of matching scores for each candidate engineer to handle task t_k. Finally, we assign task t_k to the engineer $\hat{e}$ who has the highest matching score.

$$\hat{e} = \arg \max_{(e_i, \mu_{i,k}) \in \mathcal{M}} \mu_{i,k}. \quad (11)$$

The entire architecture of *CoFa* is shown in Algorithm 1. The design of *CoFa* models the failure characteristics for both engineers and tasks, reducing noise and enhancing the accuracy of the subsequent matching steps for optimized task assignment.

Algorithm 1 The *CoFa* Framework

Input: New-received task set $\mathcal{V}'_{task}$; Defect-aware graph $\mathcal{G}$.
Output: Optimal assignment plan $\mathcal{A}$

$\mathcal{A} \leftarrow \emptyset$
`// Defect-Aware Representation`
for each engineer e_i **do**
  Formulate the engineer e_i's normalized representation $\bar{\mathbf{e}}_i$
end for
for each new-received task $t_k \in \mathcal{V}'_{task}$ **do**
  Obtain task normalized representation $\bar{\mathbf{t}}_k$ based on the embedding results of DeepWalk on $\mathcal{G}$
  `// Dynamic Candidate Selection`
  Formulate engineer candidates E_{cand} based on representations $\bar{\mathbf{e}}_i$ of each engineer e_i and the new-received task $\bar{\mathbf{t}}_k$
  `// Contrasting-Based Assignment`
  $\mathcal{M} \leftarrow \emptyset$
  for each engineer $e_i \in E_{cand}$ **do**
    Estimate the matching score $\mu_{i,k}$ of e_i to handle task t_k
    $\mathcal{M} \leftarrow \mathcal{M} \cup \{(e_i, \mu_{i,k})\}$
  end for
  $\hat{e} = \arg\max_{(e_j, \mu_{j,k}) \in \mathcal{M}} \mu_{j,k}$ ▷ Select the most appropriate engineer.
  $\mathcal{A} \leftarrow \mathcal{A} \cup \{(t_k, \hat{e})\}$ ▷ Determine the assignment.
end for
return $\mathcal{A}$

IV. Experiments

A. Experimental Setup

add more details of simulation

Dataset and Preprocessing. In this study, we apply a real-world dataset collected from a semiconductor company as mentioned in Sec. II-A. To simulate a more workload-intensive real-world scenario, we expanded the number of engineers to 26, increased the defect types from 8 to 14, and scaled up the number of tasks to 3,190 to simulate work congestion. We used the first 75% of the total RMA count as training data and the remaining 25% are used as testing data for our framework. To allocate the number of days an engineer takes to handle an RMA, we used a sigmoid function and simulated the number of days based on the engineer's familiarity with the specific type of defect. The handling proficiency was adjusted not only by the number of times an engineer dealt with that type and their average processing time but also by the current workload. The statistics of the original dataset and the simulated dataset are shown in Table II.

TABLE II: Dataset statistics.

Datasets	# Testers	#Test items	#Defect types	# Training ATE logs	# Testing ATE logs	Avg. handling days
ATE$_{\text{orig}}$	4	1,214	8	92	31	38.3
ATE$_{\text{exp}}$	26	1,803	14	2,391	799	41.1

Baselines. To demonstrate the effectiveness of the our method, we compare *CoFa* with the baselines, which can be divided into three main groups: (i) methods considering only the engineer's workload, (ii) methods considering only the user's expertise, and (iii) ablation cross-comparison. From the experimental results, we can see that our method achieves a balance in task success rate, diversity, and engineer work load.

- **LowestWL**: This method assigns the newly received task to the engineer who currently has the lowest workload.
- **MostEXP**: This method identifies the engineer with the highest count of tasks handled for the predicted task type, treating this engineer as the most experienced and assigning tasks accordingly.
- **CM-S**: CM-S is a variant of our *CoFa* that removes the *dynamic candidate selection* phase.
- **CM-SC**: CM-SC is a variant of our *CoFa* that removes both the *dynamic candidate selection* phase and the *contrasting-based assignment* phase. Instead, the assignment is determined by simply selecting the engineer whose representation has the highest cosine similarity to the representation of the new task.
- **CM-G**: CM-G is a variant of our *CoFa* that removes the embedding of the defect-aware graph. Instead, the representation of each new task is formulated based on the cosine similarity between the fail rates of each test item in the ATE logs of the new task and the fail rates of each item in each defect type.

The LowestWL and MostEXP only consider task requirements and specific information about engineers. The methods CM-S, CM-SC, and CM-G are designed as variants of our methods for the ablation study.

Evaluation Metrics. To quantitatively evaluate our proposed *CoFa* framework, we assessed the task allocation results in the test dataset. In our work, we use *Success Rate*, *Diversity*, and *Workload* to evaluate the effectiveness and balance of task allocation, reflecting the performance of task distribution to engineers.

- **Success Rate**: A task is considered successful if its completion time is earlier than the close date of an RMA task in the dataset (variable $rma.close$ in the Table I). This measure allows us to compare whether the completion time is effectively shortened compared to other baselines.
- **Diversity**: After obtaining the number of tasks handled by each engineer for all types, we calculate the standard

TABLE III: Performance comparison on dataset $\mathbf{ATE}_{\text{orig}}$. The best success rate results are highlighted in **bold**. Cells in gray indicate performance above the average value.

Metric	LowestWL	MostEXP	CM-S	CM-SC	CM-G	***CoFa***
Success Rate (↑)	77.41%	25.81%	74.19%	80.66%	83.87%	**87.10%**
Diversity (↑)	3.1728	3.5068	2.9870	3.3584	3.2278	3.0423
Workload (↓)	3.3915	1.9110	1.4407	1.7114	1.5248	1.6452

TABLE IV: Performance comparison on dataset $\mathbf{ATE}_{\text{exp}}$. The best success rate results are highlighted in **bold**. Cells in gray indicate performance above the average value.

Metric	LowestWL	MostEXP	CM-S	CM-SC	CM-G	***CoFa***
Success Rate (↑)	78.01%	26.05%	78.53%	50.13%	77.75%	**82.46%**
Diversity (↑)	8.1335	10.5638	8.0312	9.7354	8.1639	8.3245
Workload (↓)	0.6557	1.2555	0.6409	1.0883	0.5553	0.6783

deviation of the number of tasks handled by each engineer and take the average value. This result reflects the average work diversity.

- **Workload**: We calculate the current daily workload of each engineer, divide it by the workload limit, and average it over the number of days. This gives us the average workload ratio, representing the average load level of all engineers.

Implementation details. For the default settings of the extended DeepWalk model, we set the embedding dimension to 256, the walk length to 20, the number of walks to 10, and the training epoch to 50. For the hyperparameter settings, ϑ (Eq. 7) is set to 0.7, and γ (Eq. 10) is set to 2. Our experiments are performed on a system equipped with 12 CPU cores and 64GB RAM, running on CUDA version 12.1.

B. Results and Analysis

Comparison Results. Table III and Table IV show the comparison results of *CoFa* and baselines. The following observations are drawn:

- In both datasets, our method *CoFa* achieves the highest success rate among all methods. Although our diversity and workload metrics are not the lowest, their performance is still above average and very close to the best values. This result demonstrates that *CoFa* can achieve the highest task handling success rate before the close date of the task in the optimized task assignment problem. At the same time, it prevents excessive engineer workload and allows engineers to handle different types of defect types of RMA tasks, helping them accumulate experience.
- Additionally, it can be observed that the success rates of our method's variants (CM-S, CM-SC, and CM-G) are all lower than *CoFa*. Although CM-S shows better diversity and workload values than *CoFa* in both datasets, its success rate is approximately 4% lower in $\mathbf{ATE}_{\text{exp}}$ and over 10% lower in $\mathbf{ATE}_{\text{orig}}$. This result highlights the importance of each component of our method and the necessity of selecting candidate engineers based on the divergence similarity of defect-aware representations.

TABLE V: The performance of *CoFa* with different value of ϑ on dataset $\mathbf{ATE}_{\text{orig}}$. The best results are highlighted in **bold**.

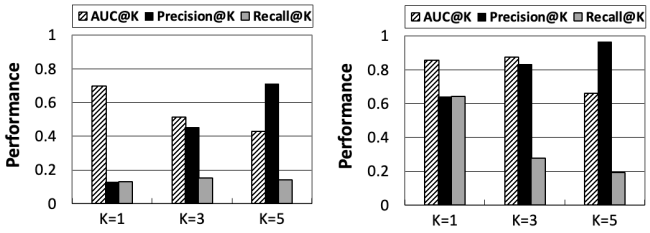
Metric	$\vartheta = 0.3$	$\vartheta = 0.5$	$\vartheta = 0.7$
Success Rate (↑)	83.87%	67.74%	**87.10%**
Diversity (↑)	**2.9612**	3.2730	3.0423
Workload (↓)	**1.4043**	1.6096	1.6452

TABLE VI: The performance of *CoFa* with different value of ϑ on dataset $\mathbf{ATE}_{\text{exp}}$. The best results are highlighted in **bold**.

Metric	$\vartheta = 0.3$	$\vartheta = 0.5$	$\vartheta = 0.7$
Success Rate (↑)	27.88%	67.15%	**82.46%**
Diversity (↑)	**8.2418**	9.4000	8.3245
Workload (↓)	1.1738	0.8239	**0.6783**

Effectiveness of Dynamic Selection. In Table III and Table IV, the importance of considering the dynamic candidate selection phase is demonstrated. To further evaluate the impact of this phase on performance, we vary ϑ in Eq. 7 which controls the similarity of the selected candidates. We vary the value ϑ across $\{0.3, 0.5, 0.7\}$. A smaller ϑ value indicates that the selected candidates have more experience in handling specific failure characteristics of the task. The experimental results are revealed in Table V and Table VI.

From the experimental results, it can be observed that when $\vartheta = 0.7$, the success rate is highest for both datasets. Conversely, when $\vartheta = 0.3$, the success rate shows significant variation in Table V and Table VI. This discrepancy may be attributed to the fewer RMA log tasks in the $\mathbf{ATE}_{\text{orig}}$ dataset, where assigning new tasks to engineers with more similar handling experience does not significantly impact their workload. However, in the $\mathbf{ATE}_{\text{exp}}$ dataset, which has a much larger number of RMA log tasks, choosing engineers with similar handling experience when $\vartheta = 0.3$ can lead to



(a) Dataset $\mathbf{ATE}_{\text{orig}}$. (b) Dataset $\mathbf{ATE}_{\text{exp}}$.

Fig. 3: The performance of defect type detection using *CoFa* across different values of K.

excessive workloads for specific engineers, thereby slowing down task processing and reducing the success rate.

Performance on Defect Type Detection. Besides verifying the effectiveness of our method in the optimized task assignment problem, we also validate the performance of our designed defect-aware graph and the extended DeepWalk embedding method for defect type detection. We simply base this on the normalized defect-aware representations of each new RMA task (Eq. 6). For defect type detection, we select the top K features corresponding to each dimension (defect type) of the representation as the detection results.

- Fig. 3 shows the performance of $AUC@K$, $Precision@K$, and $Recall@K$ when varying $K = \{1, 3, 5\}$. As expected, when the value of K increases, the precision also increases. Notably, in both datasets, the highest AUC values are obtained when $K = 1$. This result indicates that our designed extended DeepWalk, which considers second-order proximity during training, helps improve the performance of defect type detection.
- Fig. 4 and Fig. 5 show the defect type detection performance for each value of K under different walk lengths in the extended DeepWalk model. Most of the results indicate that performance does not significantly change with varying walk lengths. However, it is notable that in Fig. 4, when the walk length exceeds 20, the AUC values for different K values show improvement. This is because, with longer walk lengths, the consideration of second-order proximity in DeepWalk training becomes more pronounced, leading to greater proximity distinction. Consequently, this allows the model to more accurately determine the probability of each RMA task belonging to different defect types based on the graph structure.

V. Related Work

A. Fault Localization

Research on IC testing primarily focuses on optimizing test design in hardware design and manufacturing processes. Fan *et al.* [3] integrated effective data-driven approaches and machine learning methods for equipment monitoring, fault detection, and diagnosis during the thin film deposition stage. *et al.* [4] addressed data imbalance issues in the thin film deposition process by using VAE-based generative models for data augmentation, particularly when defect data is scarce, thus improving detection and classification. He *et al.* [5] modeled each step of the etching process to achieve subsystem-level detection and classification, enhancing equipment utilization and production quality. Lang *et al.* [6] employed Kernel Density Estimation methods to directly use known successful operational data for training during the plasma etch and ion implantation stages, achieving effective anomaly detection through single-class anomaly detection without fault data. Liau *et al.* [12] focused on detecting and analyzing power supply noise using automated test equipment (ATE), utilizing neural networks and genetic algorithms for learning and classification to identify IC power noise issues.

B. Task Assignment

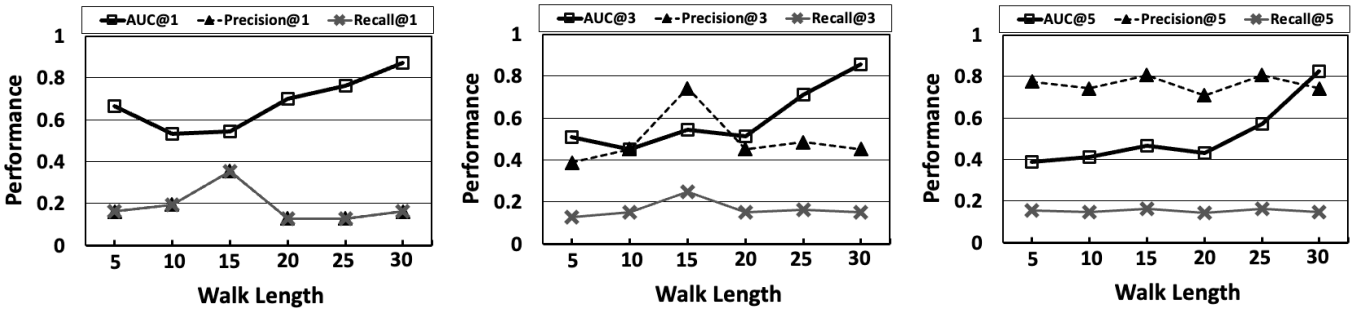
In task assignment research, Ho *et al.* [13] proposed a method that considers worker skills and task requirements to ensure task quality and maximize worker utilization. Zheng *et al.* [14] developed the QASCA system, focusing on quality-aware task assignment and emphasizing the importance of considering worker skills and quality standards in task allocation. Mo *et al.* [15] optimized majority voting mechanisms for human intelligence tasks to improve task assignment efficiency and accuracy. Assadi *et al.* [13] emphasized effective allocation in heterogeneous task environments with their online heterogeneous task assignment method. Task assignment also includes various evaluation metrics, such as those used by Li *et al.* [16] to identify high-quality worker groups and allocate tasks to them. Singer *et al.* [17] adjusted prices based on task difficulty and worker quality to evaluate the cost-effectiveness of task completion. Mavridis *et al.* [18] assessed the time required for workers to complete tasks, ensuring timely task completion. Goel *et al.* [19] considered fairness among the population under budget constraints.

VI. Conclusion

In this paper, we investigate how to model the failure characteristics of RMA tasks and determine the most appropriate engineer to handle these tasks. We develop *CoFa*, a framework that learns defect-aware representations for engineers and RMA tasks based on a designed defect-aware graph and an extended embedding model. Furthermore, *CoFa* utilizes a dynamic selection process to determine the engineer candidates who may understand similar failure characteristics of tasks and employs a contrasting-based method to complete the assignment. Applied to real-world IC testing data, *CoFa* achieves the highest success rates while balancing the workloads and task diversity handled by engineers. The experimental results further demonstrate the practicality of our method for detecting the defect types of ICs.

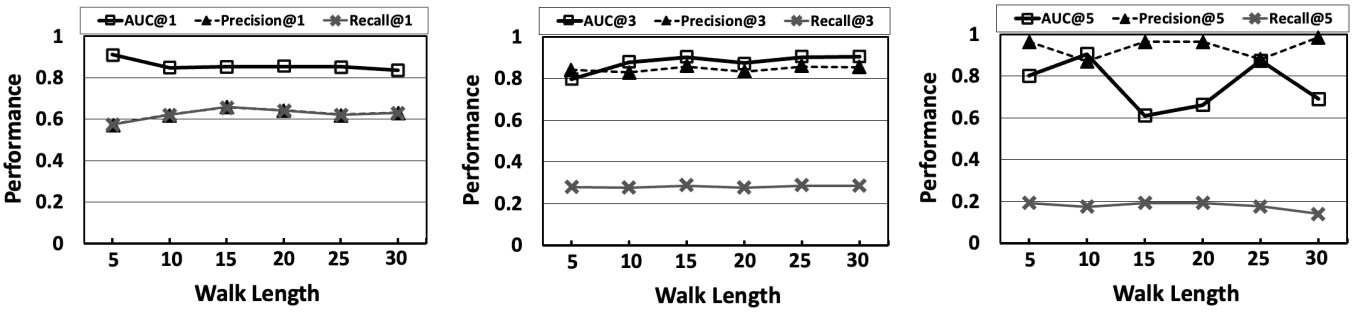
References

[1] S. K. K, S. S. Rekha, A. Koushik, A. K. Swain, and K. K. Mahapatra, "Lightweight secured split test technique with rma capability to prevent ic counterfeiting," in *2023 IEEE International Symposium on Smart Electronic Systems (iSES)*, 2023, pp. 191–196.



(a) $K = 1$. (b) $K = 3$. (c) $K = 5$.

Fig. 4: The performance of defect type detection using *CoFa* across different walk lengths on the dataset $\mathbf{ATE}_{\text{orig}}$.



(a) $K = 1$. (b) $K = 3$. (c) $K = 5$.

Fig. 5: The performance of defect type detection using *CoFa* across different walk lengths on the dataset $\mathbf{ATE}_{\text{exp}}$.

[2] Y. Lee, T. Kane, J.-J. Lim, L. Schiano, Y.-B. Kim, F. Meyer, F. Lombardi, and S. Max, "Analysis and measurement of timing jitter induced by radiated emi noise in automatic test equipment," *IEEE Transactions on Instrumentation and Measurement*, vol. 52, no. 6, pp. 1749–1755, 2003.

[3] S.-K. S. Fan, C.-Y. Hsu, D.-M. Tsai, F. He, and C.-C. Cheng, "Data-driven approach for fault detection and diagnostic in semiconductor manufacturing," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 4, pp. 1925–1936, 2020.

[4] S.-K. S. Fan, D.-M. Tsai, and P.-C. Yeh, "Effective variational-autoencoder-based generative models for highly imbalanced fault detection data in semiconductor manufacturing," *IEEE Transactions on Semiconductor Manufacturing*, vol. 36, no. 2, pp. 205–214, 2023.

[5] Q. P. He and J. Wang, "Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 20, no. 4, pp. 345–354, 2007.

[6] C. I. Lang, F.-K. Sun, B. Lawler, J. Dillon, A. A. Dujaili, J. Ruth, P. Cardillo, P. Alfred, A. Bowers, A. Mckiernan, and D. S. Boning, "One class process anomaly detection using kernel density estimation methods," *IEEE Transactions on Semiconductor Manufacturing*, vol. 35, no. 3, pp. 457–469, 2022.

[7] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.

[8] A. Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in *Proc. of ACM SIGKDD*, 2016.

[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. of Workshop at ICLR*, 2013.

[10] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," *Proceedings of the 24th International Conference on World Wide Web*, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:8399404

[11] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. of International Conference on Advances in neural information processing systems*, ser. NIPS '13, 2013, pp. 3111–3119.

[12] E. Liau and D. Schmitt-Landsiedel, "Evolution of automatic semiconductor test equipment: automatic test pattern learning, classification, optimisation and generation for power supply noise," in *IEEE International Symposium on Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2003. VECIMS '03. 2003*, 2003, pp. 39–44.

[13] C.-J. Ho and J. W. Vaughan, "Online task assignment in crowdsourcing markets," in *AAAI Conference on Artificial Intelligence*, 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:1145053

[14] Y. Zheng, J. Wang, G. Li, R. Cheng, and J. Feng, "Qasca: A quality-aware task assignment system for crowdsourcing applications," *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:207223616

[15] L. Mo, R. Cheng, B. Kao, X. S. Yang, C. Ren, S. Lei, D. W.-L. Cheung, and E. Lo, "Optimizing plurality for human intelligence tasks," *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:13568616

[16] H. Li, B. Zhao, and A. Fuxman, "The wisdom of minority: discovering and targeting the right group of workers for crowdsourcing," *Proceedings of the 23rd international conference on World wide web*, 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:13803906

[17] Y. Singer and M. Mittal, "Pricing mechanisms for crowdsourcing markets," *Proceedings of the 22nd international conference on World Wide Web*, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:10536454

[18] P. Mavridis, D. Gross-Amblard, and Z. Miklós, "Using hierarchical skills for optimized task assignment in knowledge-intensive crowdsourcing," *Proceedings of the 25th International Conference on World Wide Web*, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:19355360

[19] N. Goel and B. Faltings, "Crowdsourcing with fairness, diversity and budget constraints," in *Proceedings of the 2019 AAAI/ACM Conference*

on AI, Ethics, and Society, ser. AIES '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 297–304. [Online]. Available: https://doi.org/10.1145/3306618.3314282